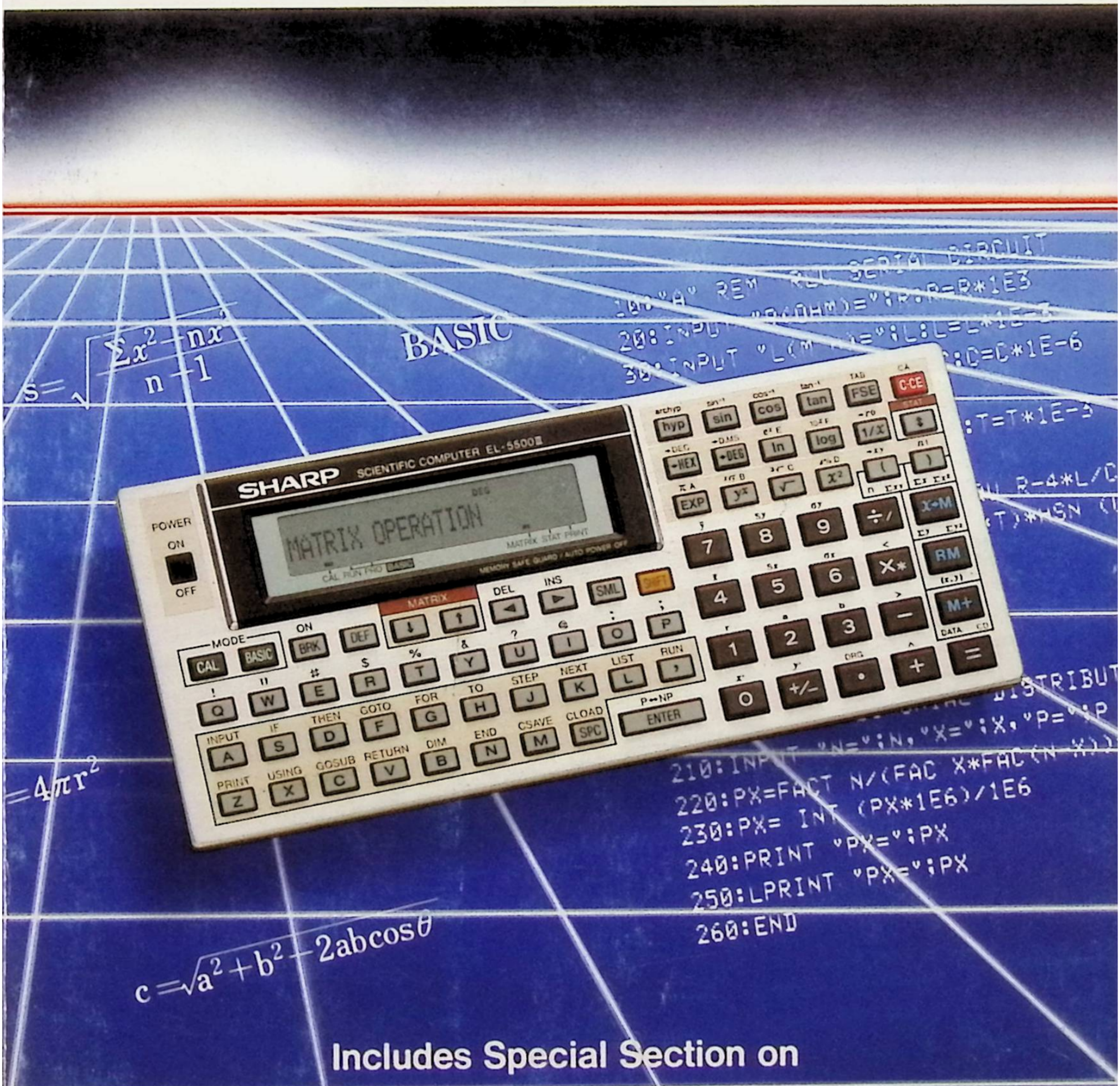
COMPUTING THE SCIENCES

Applications for the SHARP EL-5500III Scientific Computer



Includes Special Section on

BUSINESS and FINANCE

COMPUTING THE SCIENCES

Applications for the SHARP EL-5500III Scientific Computer

TABLE OF CONTENTS

NOTICE

Some answers in this text have been condensed. Your results may display more digits than those in the text. Please refer to your operation manual for calculation ranges.

CHAPTER 1

INTRODUCTION TO PROGRAMMING

Programming may be thought of as a series of steps. These are: analysis of the problem, preparing an algorithm, flowcharting, writing the program, testing and editing and documentation. Each of these steps is performed by the programmer. These steps apply to all programmable machines from calculators to complex computers.

STEP 1: Analyzing the Problem

A well defined problem has a clear precise statement of the situation with a recognizable end result. If the problem is not well defined, developing a solution is extremely difficult.

Example of a well defined problem:
The radius of a circle is 10 inches. Determine the area of a circle.

STEP 2: Algorithms

Once the problem has been clearly defined, a method of solution must be determined. The desired result is obtained by establishing a series of events called an algorithm.

The computer can only perform a single operation at a time. We are forced to think in sequential terms because of the nature of the computer. From the computer's point of view, the algorithm is the order in which it must perform a set of operations.

The basic formula for the problem in Step 1 is: Area of a circle=Radius squared multiplied by constant Pi.
The algorithm for this problem is:

A. Take the value of the radius

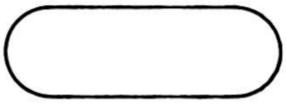
B. Square this value

C. Multiply the result of "B" by Pi

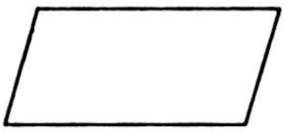
D. Display the result

STEP 3: Flowcharting

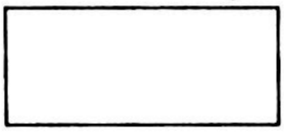
The logic of the algorithm can be visually presented and checked by the use of a flowchart. Each type of operation in the algorithm has a distinct shape. The basic shapes that apply to the handheld computer are:



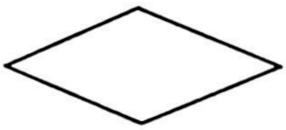
A. Oval — Signifies the beginning and ending of a program



B. Parallelogram — Input and output of data

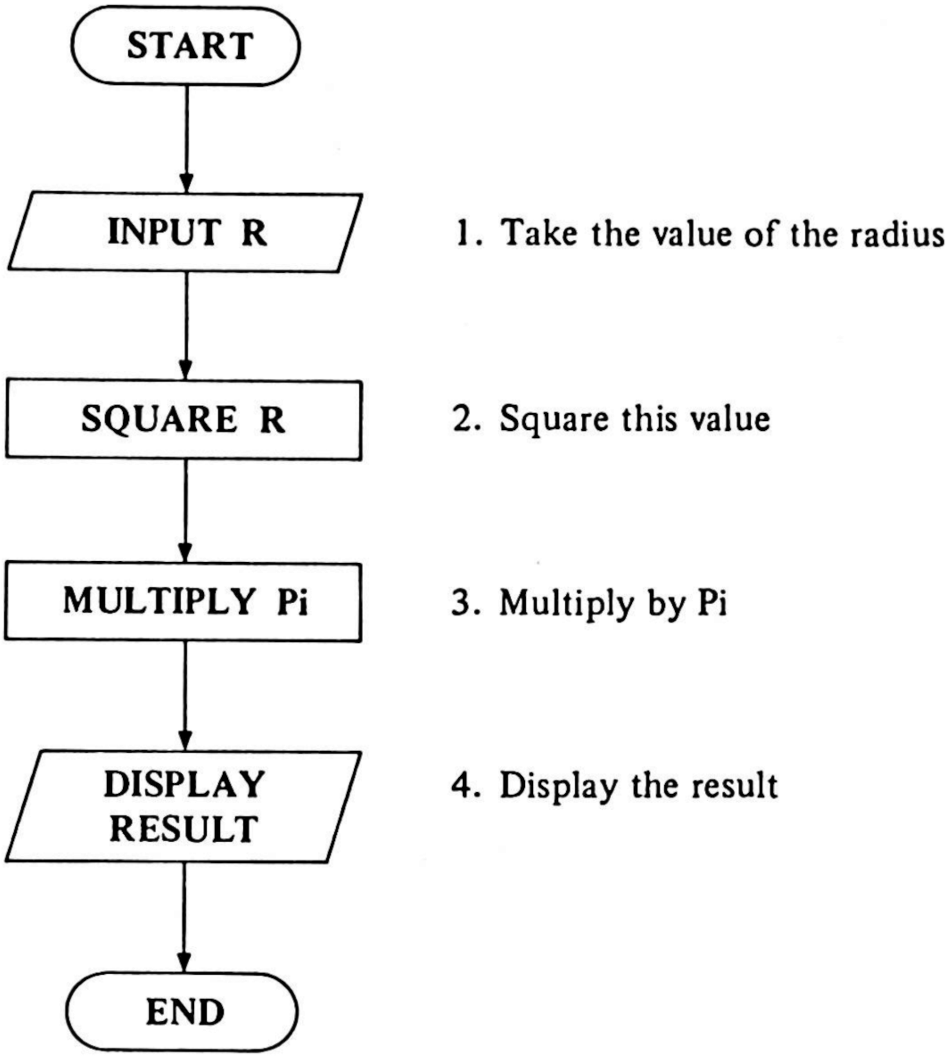


C. Rectangle — An operation (multiplying, division, etc.) will be carried out



D. Rhombus — A decision will be made

Flowcharting the problem of the area of a circle:



STEP 4: Programming

The next step is to write the actual program. On the EL-5500III, programming is done in the BASIC language. BASIC is the most popular of the computer languages and is available and relatively easy to learn. Detailed discussion of the EL-5500III's BASIC is the subject of Chapter 3.

STEP 5: Testing and editing

Use data with a known answer to the problem to check a program. The flowchart provides a means of determining where error may have occurred. Errors in programming technique will result in an error code. Each error message is

described in detail in Chapter 3.

STEP 6: Documentation

A formal method of keeping program records should be developed. Many problems will have more than one variable. The order of entry of each item in a set of data must be specified if the program does not ask for each data point by name or symbol.

The program's documentation consists of the flowchart, the program, commentaries for future reference, and user instructions.

CHAPTER 2

USING THE EL-5500III AS A CALCULATOR

The EL-5500II operates as a calculator in the **CAL** mode. Calculations may also be performed without programming in the **RUN** mode under **BASIC**. The two types of calculating are described in Parts 1 and 2 of this chapter.

PART 1

CAL Mode

$\boxed{\text{C·CE}}$ Clear Entry

An incorrectly entered number can be replaced as long as the number has not already been followed by a "function key." Pressing the clear entry key will clear the latest entry while retaining all previous entries; for example:

Key in: 5 **x** 4 (The 4 should be 6)

Key in: **C·CE** 6 =

Answer: 30

If the C·CE key is pressed twice, the calculator will be completely cleared of the current calculation. All previous calculations will be cleared if the C·CE key is pressed after a function key.

+ – x ÷ = **Addition, Subtraction, Multiplication, Division, Equals**

Calculate: 5 + 2 x 3 – 2 ÷ 0.5

Key in: 5 **+** 2 **x** 3 **–** 2 **÷** .5 **=**

Answer: 7

A lot of land is 100 feet by 200 feet. An acre is 43,560 square feet. How many

acres in the lot?

Key in: 100 x 200 ÷ 43560 =

Answer: 0.46 acres (approximately 1/2 acre)

There is a cost to not taking advantage of cash discount terms. The cost as an equivalent annualized interest charge on the purchase is:

Annualized Interest Rate (%)

= (discount percent)/ [100 – (discount %)] x

365/ [(payment days) – (discount period)] x 100

Terms are 2%/10 days, net 30; actual payment is made in 30 days. What is the interest cost to the buyer? These terms mean that the buyer receives a 2% discount from the purchase price if payment is made within ten days of receiving the bill. The 2% discount is not allowed if payment is made after ten days.

2/(100 – 2) x 365/(30 – 10) x 100 =

2/98 x 365/20 x 100

Key in: 2 ÷ 98 x 365 ÷ 20 x 100 =

Answer: 37.24%

Pressing the = key gives the answer to the entered formula. Press = again and the last entry and function will be carried out.

Calculate: 4 + 2 + 2 + 2

Key in:	*Answer:*
4 + 2 =	6
=	8
=	10

Calculate: 1/2 of 1/2 of 1/2

Key in:	*Answer:*
1 ÷ 2 =	0.5
=	0.25
=	0.125

The number after the operation automatically becomes the constant in constant calculations, except in multiplication. In multiplication, the first entry becomes the constant.

Calculate the values of each digit in the octal numbering system (base 8). Each position is a multiple of 8.

Key in:	*Answer:*
8 x 1 =	8
=	64
=	512
=	4096

The distance from New York to San Francisco (SF) is 2,572 miles; from SF to Bangkok, 7,931; SF to Hong Kong, 6,905; SF to Peking, 5,918; and SF to Singapore, 8,448. What is the distance to each of the cities from New York via SF?

Key in:

	Constant		*Answer:*	New York to:
7931 +	2572	=	10503	Bangkok
6905		=	9477	Hong Kong
5918		=	8490	Peking
8448		=	11020	Singapore

The weights of three overweight individuals are 217, 310, and 343 pounds. Under a weight control program, they each agree to loss a pound a day for the first month. What are their weights at the end of the month? If the heaviest agreed to lose twice as much during the month, what is his weights?

Key in:

Weight		**Constant**		*Answer*	
217	−	30	=	187	
310			=	280	
343			=	313	
			=	283	(double deduction)

The price of gasoline is \$1.20 per gallon. A car averages 30 miles per gallon. What would be the gasoline cost for trips of 400, 1000, and 2500 miles?

Note: The first-calculaition establishes the constant.

Key in:

Constant:				*Answer:*
1.2 ÷ 30	×	400	=	16
		1000	=	40
		2500	=	100

Trade or chain discounts are a series of discounts deducted successively from the list price. For example: A company provides discounts of 30%, 5% and 2%. What is the final price on products having list prices of: \$200, \$100 and \$50?

Note: Calculate the full discount by using an equivalent single discount which is the resultant of each discount multiplied by the other.

Single Discount Equivalent: $(1-r_1)\ (1-r_2)\ (1-r_3)$

$$(1-0.3)\ (1-0.05)\ (1-0.02)$$

$$0.7 \times 0.95 \times 0.98$$

Key in:

							Answer:	
.7	×	.95	×	.98	×		.6517	equivalent discount
					200	=	130.34	
					100	=	65.17	
					50	=	32.59	

The constant divisor helps create a table of index numbers. Pig iron production (thousand tons) in the United States over a five year period was: 1974: 95,909; 1975: 101,208; 1976: 86,870; 1977: 81,328; 1978: 87,679. The 1974 quantity becomes the constant divisor.

Key in:		Constant:		Index #	Year
				1.00	1974
101208	÷	95909	=	1.06	1975
86870			=	0.91	1976
81328			=	0.85	1977
87679			=	0.91	1978

Use of Parentheses and Priority Levels

Mathematical operations within a single problem are carried out according to a preestablished priority. The priority of operations may be thought of as levels. The operations on the first level are single-variable functions and they will be calculated before two-variable functions on the second level. The operations on the second level will precede those on the third, fourth, and fifth levels. The operations that occur on each level are the following:

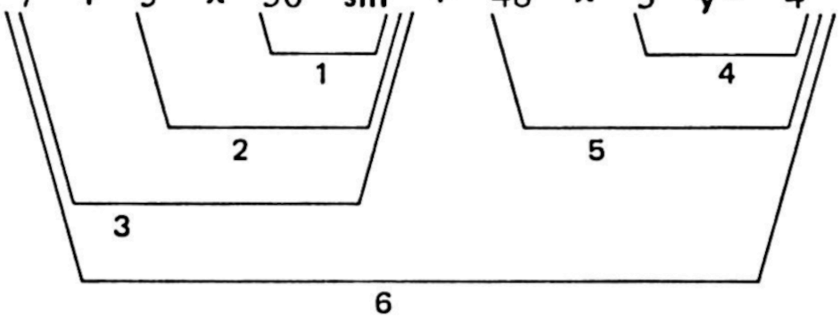
Level	Operations
(1)	Single-variable functions which are calculated as entered like sin, ln, 10^x, $1/x$, x^2
(2)	y^x, $\sqrt[x]{y}$
(3)	×, ÷
(4)	+, −
(5)	=, Δ% , M+

To demonstrate the priority levels some keys explained in later sections will be used. The numbers under the brackets indicate the sequence in which the calculations are carried out.

Calculate:

$7 + 5 \times \sin 30 + 48 \times 3^4$

Key in: 7 + 5 × 30 sin + 48 × 3 y^x 4 =



Answer: 3897.5

( ) ( ) Parenthesis Keys

The parentheses keys are needed to cluster together a series of operations when it is necessary to override the priority system of algebra. The use of parentheses is required to solve: $\dfrac{3 + 5 + 2}{7 + 9 + 4}$

Key in: (3 + 5 + 2) ÷ (7 + 9 + 4 [a]) =

Answer: 0.5

The parenthesis marked "a" may be left out and the equals key will complete all pending operations. When parentheses are in use, the symbol ( ) will appear in the display. A pending operation is an operation that has not been completed. Up to eight levels of pending operations are possible when parentheses are used.

Calculations in parentheses have priority over other calculations. Parentheses can be used up to three times in a single level. Calculations within the innermost set of parentheses will be calculated first.

Calculate: $8 \times \dfrac{2 \times (5 + 3 \times 2)}{40} - 11 \times \dfrac{16 + 4}{5 \times 8}$

Key in: 8 × ((2 × (5 + 3 × 2)) ÷ 40) − 11 ×
((16 + 4) ÷ (5 × 8) =

Answer: −1.1

EXP Exponential Key

Scientific notation is used to express numbers which are too large to write out and therefore inconvenient to use. It also makes operations easier to perform. A number (P) is written in scientific notation by dividing it into two parts: the mantissa (a) and a power of 10 with exponent (k). Therefore, the equation for any number is:

$$P = a \times 10^k$$

where k is an integer and $1 \leqq a < 10$.

Scientific notation is particularly important when using the scientific calculator. The display may be too small for a particular number. Scientific notation allows the expression of numbers which are too great or too small to fit on the ordinary display. The mantissa a in scientific notation cannot exceed ten numerals. The exponent k may only contain two digits. If more than two are entered, only the last two digits entered will be accepted. The power range is between 10^{-99} and 10^{+99}. Numbers given in scientific notation can be entered directly:

Number		Enter	Key in
798,520	=	7.9852×10^5	7.9852 **EXP** 5
9,999 (96 0's)	=	9.999×10^{99}	9.999 **EXP** 99
0.0000000000035194	=	3.5194×10^{-12}	3.5194 **EXP** 12 +/–

Large numbers with zeros can be entered in "shorthand." If the number begins with a 1 followed by zeros, only **EXP** and the power need be pressed.

Number	Enter	Key in	Display
1,000,000	10^6	EXP 6	1. E 06
80,000	8×10^4	8 EXP 4	8. E 04
.001	10^{-3}	EXP 3 +/–	1. E–03
.0000000009	9×10^{-10}	9 EXP 10 +/–	9. E–10

To state these numbers:

10^6 = "Ten to the sixth power"

8×10^4 = "Eight times ten to the fourth power"
10^{-3} = "Ten to the minus third power"
9×10^{-10} = "Nine times ten to the minus tenth power"

In many practical applications a number will be placed in the calculator with an exponent that is an increment of three; for example, enter the following numbers in terms of meters:

Number	Enter	Key in
6 kilometers	6×10^3	6 EXP 3
2.95 millimeters	2.95×10^{-3}	2.95 EXP 3 +/–
1 micrometer	10^{-6}	EXP 6 +/–

If the result of a calculation exceeds a quantity which can be shown on the display in regular notation, the calculator will automatically convert to scientific notation.

Calculate: 2 million × 5 million

Key in: 2000000 × 5000000 =

Answer: 1×10^{13}

F↔E Floating↔Exponential Key

The **E** stands for exponental or scientific notation and the **F** for floating decimal or regular notation. A number in the display which is the result of a calculation can be changed from one type of notation to the other and vice versa. If the number of integer digits will exceed ten digits in regular notation, it cannot be converted from scientific notation. To be convertible from one notation to another the value of x must be such that $0.000000001 < |x| < 9999999999$.

1/x Reciprocal Key

The reciprocal can be thought of as 1 divided by a number. The reciprocal of 5 is 1 divided by 5 or 0.2. If 0.2 is in the display and **1/x** is pressed, the answer

will be 5.

Calculate: 1/3 + 1/6

Key in: 3 **1/*x*** + 6 **1/*x*** =

Answer: 0.5 To get the fraction equivalent press **1/*x***.

Display: 2 *Answer:* 1/2

A card dealer specifies the names of three cards to be drawn in order from a deck of 52 without replacement. What are the chances that these cards will be drawn? The chance of obtaining the first is one over the total; the second is one over the remaining total; etc. The chance of a series of specified occurrences happening is each times the other or:

$$1/N \times 1/(N-1) \times 1/(N-2) \times \cdots \times 1/(N-n)$$

where, N = total number and n = number of occurrences.

solution: 1/52 × 1/51 × 1/50

The reciprocal key eliminates the need to key in "one divided by." After the answer is obtained, the reciprocal key allows the user to obtain the probabilities as they are usually stated.

Key in: 52 **1/*x*** × 51 **1/*x*** × 50 **1/*x*** =

Answer: 0.000007541

Key in: **1/*x***

Answer: 1 out of 132600

A die has 6 sides and in the throw of a pair of dice there are 36 (6 x 6) possible permutations. There are four different permutations of the two dice that add up to 5. What is the probability of getting a 5 when the dice are thrown?

Key in: 4 ÷ 36 =

Answer: 0.11111

Key in: **1/x**

Answer: 1 out of 9

TAB Tab Key

The **TAB** key is used to specify the number of decimal digits desired in the calculation result. The number of places after the decimal point is specified by a numeral key pressed after the **TAB** key. For floating decimal calculation press the key (Decimal point key) after **TAB**. Carry over will be automatically rounded:

Key in: **SHIFT TAB .** 1.234567891 **×** 1 **=**

SHIFT TAB 3 display reads 1.235

SHIFT TAB 9 display reads 1.234567891

indicating that when **TAB** 0 to 8 is used, the digits that have been rounded off can be recalled by using **TAB** with a higher number or **.** . Internal calculations are carried out using the full 12 digits to maintain accuracy.

x→M X To Memory Key

The memory is a register where a number can be stored for later use. The x-to-memory key takes the number in the display and places it in the memory, replacing whatever number was previously in memory. Pressing the clear key before this key places 0 in the memory, and the **M** symbol disappears from the display. The memory should be cleared before starting a new calculation that will use the memory. Alternatively, using the **x→M** key after the first calculation of a new problem will make it unnecessary to clear the memory.

M+ Memory Plus

Pressing this key will add the number in the display (which may be the result of a calculation) to the total in the independent memory. To subtract the value in the display from the memory, press the **+/−** key before pressing **M+**.

RM Recall Memory

The recall memory key brings back a value for later use, even when other calculations are performed in the interim. The value stored in the memory is not lost even when the calculator is turned off.

Memory Examples:

1. The inventory of a product at five different locations is 233, 280, 342, 416, and 245. During the month withdrawsls are made of 300 units and 200 units. Incoming shipments are 150, 115, and 80. What is the inventory at the beginning and at the end of the month?

Key in: 233 + 280 + 342 + 416 + 245 = x→M

Answer: 1516, beginning inventory

Key in: 300 + 200 = +/– M+
150 + 115 + 80 = M+
RM

Answer: 1361, end-of-month inventory

2. On a trip to the supermarket an individual has $50 to spend. In order to avoid overspending, memory is used to indicate the amount of money remaining. Purchases are: $6.95, 3 items at $4.95, 12 at $0.60 and 15 at $0.97.

Key in: Set **TAB** at 2.

50 x→M 6.95 + 3 × 4.95 + 12 × .6 + 15 × .97 =
+/– M+ RM

Answer: $6.45 remaining

3. Balancing The Checkbook

A running checkbook balance is kept as follows: **TAB** 2

Check #	Item	Amount	Deposit	Balance
	Initial Balance			$1123.42
722	N.E.Telephone	22.82		1100.60
723	B. Edison	18.55		1082.05
724	Mortgage Bank	651.75	782.45	1212.75
725	United Fund	50.00		1162.75
726	Bookstore	3.22	29.30	1188.83
727	American Express	242.80		946.03
	Interest Received		7.85	953.88

Note: The display shows the balance after each transaction.

The amounts of the check are subtracted from the balance and deposits are added to the total. If the balance is figured once a month, the general procedure is:

Current Balance = Beginning Balance – Value of Checks Written + Value of Deposits

Key in: 1123.42 – 22.82 – 18.55 – 651.75 + 782.45 – 50 – 3.22 + 29.3 – 242.8 = x → M

Answer: $946.03 balance

The bank statement arrives and shows a total in the account of $1,007.10. The difference between the bank statement and the checkbook balance of $946.03 is due to undeposited checks and unrecorded interest received on the daily balance in the account. Checks for $50.00 and $3.22 had not been returned to the bank in time for inclusion in this statement. The checking account balance earns $7.85 interest.

Bank Balance = Current Balance (A) + Value of Undeposited Checks (B) + Value of Unrecorded Interest (C)

	A		B		C	
Key in:	RM	+ 50 +	3.22	+	7.85	=

Answer: $1,007.10 bank balance

4. Invoicing

A customer buys three items: 50 at $8.25 each, 2 at $242 each and 160 at $10.95 each. There is a 1/3 discount for total purchases exceeding $1,000, a 10% shipping charge, and 8% sales tax. What is the final bill? After clearing the memory, calculate the amount for each item and add to the memory register. The memory will have the total of all purchases. If the gross amount of the invoice exceeds $1000, calculate a a/3 discount of the amount and deduct the discount from the total which is in the memory. Recall the new amount in memory to calculate the shipping and tax charges.

Key in: 0 x→M

50 × 8.25 M+
2 × 242 M+
160 × 10.95 M+
RM ÷ 3

+/– M+ (to subtract discount from total)

RM × 10 Δ% M+
RM × 8 Δ% M+
RM

Answer: $2097.61

Δ% Delta Percent Key

Percent means to divide a quantity into 100 parts and to make comparisons on that basis. A decimal ratio multiplied by 100 is the percent. One-half or 0.5 is the same as 50%. The percent key completes an operation. Use it to find:

a. A part of a whole

b. The percent that one number represents of a total.

c. The percent change

Calculate 25% of 800:

Key in: 800 × 25 Δ%

Answer: 200

25 is 20% of what number?

Key in: 25 ÷ 20 Δ%

Answer: 125

The delta percent key can also be used to automatically calculate differences or to measure change. A sum of 1000 is what percent greater than 800? The normal method of calculating would be:

$$\frac{1000 - 800}{800} \times 10 = 25\%$$

Key in: 1000 − 800 Δ%

Answer: 25%

The retail price of a product is equal to the cost plus markup. A product has a wholesale cost of $68. What is the retail price if the markup on cost is 60%?

$$\text{Retail Price} = \text{Cost} + \text{Markup}$$
$$160\% = 100\% + 60\%$$

Key in: 68 × 160 Δ%

Answer: $108.80

If the cost is $68 and the markup is 60% based on the selling price, what is the selling price?

$$\text{Retail Price} = \text{Cost} + \text{Markup}$$
$$100\% = 40\% + 60\%$$

The cost is 40% of the retail price.

Key in: 68 ÷ 40 Δ%

Answer: $170.00

Retail Price	=	$170	=	100%
Cost	=	68	=	40%
Markup	=	102	=	60%

The markup based on cost of an item selling for $110 and costing $70 is:

$$\frac{110 - 70}{70} \times 100$$

Key in: 110 − 70 Δ%

Answer: 57.1%

The markup based on the selling price for the same item is:

$$\frac{70 - 110}{110} \times 100$$

Key in: 70 − 110 Δ%

Answer: 36.4% (The negative sign is disregarded.)

There are 150,000,000 motor vehicles in the United States. Average peak efficiency of 25 miles per gallon is obtained at a speed of 45 miles per hour. At 55 mph, efficiency is reduced by 10% and at 65 mph by 25%. (The numbers in this example are hypothetical.) Each motor vehicle averaged 3,000 miles per year that would have been travelled at 65 mph before the 55 mph speed limit was strictly enforced. How many gallons per year are saved at 55 mph? Solve for the mileage at various speeds.

Key in: 25 × 90 Δ%

Answer: 22.5 mpg at 55 mph

Key in: 25 × 75 Δ%

Answer: 18.75 mpg at 65 mph

Solve for the number of gallons used at these speeds. Pressing **EXP** 7 gives the scientific notation for seven zeros. Scientific notation can be used in business problems when large numbers are used. Record the total number of miles driven in memory for later use.

	Motor Vehicles	Miles/year					
Key in:	15 **EXP** 7 **×**	3 **EXP** 3	**=**	**x→M**	**÷**	22.5	**=**

Answer: 2×10^{10} gallons at 55 mph

Key in: **RM** **÷** 18.75 **=**

Answer: 2.4×10^{10} gallons at 65 mph

Subtract the answer at 55 mph to get gallons saved

Key in: **–** 2 **EXP** 10 **=** **F↔E**

Answer: 4×10^{9} or 4 billion gallons/year saved.

Calculate the percent of gasoline/year saved. In any calculation of ratios, like percent change, the exponential portion of a number may be left out if all the numbers have the same exponent.

Key in: 2 **–** 2.4 **Δ%**

Answer: 16.67%

[↕] Exchange Key

The last number in a calculation can be checked with the exchange key:

Key in: 30 **÷** 365 **=**

Display: 0.08219178

Key in: **↕**

Display: 365

The exchange key can be used to determine what part a number is of a whole without reentering the divisor.

The population of 3 towns is 284, 1096, and 789. What proportion of the total population of the three towns is the second town?

Key in: (284 + 1096 + 789) ÷ 1096 ↕ =

Answer: 0.5053

The exchange key is used in polar coordinate conversions as explained later in this section.

$\boxed{x^2}$ Square

The square of a number is obtained by multiplying a number (x) by itself: $x \times x = x^2$. A square is five feet on each side; what is the area?

Key in: 5 x^2

Answer: 25 square feet

Calculate: $(-25.6)^2$

Key in: 25.6 +/– x^2

Answer: 655.36

A piece of property is 350 feet on each side. How many acres is the area? Note: 1 acre = 43,560 square feet.

Key in: 350 x^2 ÷ 43560 =

Answer: 2.81 acres

$\boxed{y^x}$ Power Key

The base number is y and the exponent is x.

Calculate: 5^2, 5^3, $(22.5 + 18.6)^{7.8}$, $2^{(12.3+4.6)}$, -3^3, $\sqrt{3}$

Key in:	*Answer:*
5 y^x 2 =	25.
5 y^x 3 =	125.
(22.5 + 18.6) y^x 7.8 =	$3.872275162 \times 10^{12}$
2 y^x (12.3 + 4.6 =	122294.5003
3 x/– y^x 3 =	–27.
3 y^x 2 1/x =	1.732050808

A negative power is the same as a reciprocal:

$$4^{-3} = \left(\frac{1}{4}\right)^3 = \frac{1}{4 \times 4 \times 4} = \frac{1}{64} = 0.015625$$

Key in: 4 y^x 3 +/– =

Answer: 0.015625

[√‾] Square Root

The square root of a number is defined as the quantity which when multiplied by itself equals the original number. The original number cannot be a negative, as the answer would be imaginary and the calculator is not equipped to handle this problem. The display will always show the answer as positive, but in reality all answers are preceded by ±; the correct answer may be negative. The square root key completes an operation and the equals key is not necessary.

Calculate: the square roots of: 64, 0.25, 7.895 and 24 + 82 + 95

Key in:	*Answer:*
64 √‾	8
.25 √‾	0.5
7.895 √‾	2.809804264
24 + 82 + 95 = √‾	14.17744688

The walls of a house will be 12 feet high; if the base of a ladder is to be situated five feet from the house, what is the minimum height of a ladder to reach the roof?

A special equation which applies to all right triangles is known as the Pythagorean Theorem:

$$c^2 = a^2 + b^2$$

c is the hypotenuse of the triangle.

$$c = \sqrt{a^2 + b^2}$$

Solve for c:

Key in: $\underset{a}{5}\,x^2 + \underset{b}{12}\,x^2 = \sqrt{\ }$

Answer: 13 feet – minimum ladder height

[3√‾] Cube Root

The cube root aids in calculating a quantity which, when multiplied by itself three times, equals the original value. The original value may be negative.

A box of equal-sized sides has a volume of 27 cubic meters; how big is each side?

Key in: 27 $\sqrt[3]{\ }$

Answer: 3 meters

The cube root of −125 is:

Key in: 125 +/− $\sqrt[3]{\ }$

Answer: −5

[$^x\sqrt{y}$] Root Key

The root key calculates the xth root of y.

Calculate: $\sqrt{64}$, $\sqrt[3]{27}$, $\sqrt[0.25]{3}$, $\sqrt[5]{(25 \times 82 - 50)}$, and $\sqrt[5]{-32768}$, and $\sqrt[10]{10}$

Key in:	*Answer:*
64 $\sqrt[x]{y}$ 2 =	8
27 $\sqrt[x]{y}$ 3 =	3
3 $\sqrt[x]{y}$ .25 =	81
(25 × 82 − 50) $\sqrt[x]{y}$ 5 =	4.573050519
32768 +/− $\sqrt[x]{y}$ 5 =	−8
10 $\sqrt[x]{y}$ 10 =	1.258925412 (decibel)

$\boxed{n!}$ Factorial

Multiplying a number n by a series of numbers starting with one and incrementing by one until n has been reached is called factorial. The calculator is limited to whole numbers for n which are greater than or equal to 0. The quantity 0! is defined to have a value of one. The factorial key completes an operation and the equals key is not necessary.

$n! = n \times (n-1) \times \ldots \times 4 \times 3 \times 2 \times 1$

$4! = 4 \times 3 \times 2 \times 1$

Key in: 4 **n!**

Answer: 24

The calculated value on the calculator cannot exceed $9.99999999 \times 10^{99}$. The highest factorial that can be calculated is 69 n!.

Answer: $1.711224524 \times 10^{98}$

Try: 70 **n!**

Display: **E** (Error)

An error symbol occurs because the answer overflows the capacity of the calculator.

There are six different cards. In how many different ways or orders can any three be drawn from this group? In this case $n = 6$, $r = 3$.

$$\frac{6!}{(6-3)!} = \frac{6!}{3!}$$

Key in: 6 **n!** ÷ 3 **n!** =

Answer: 120

How many poker hands (5) in a deck (52)?

$$\frac{52!}{5!\ (52-5)!} = \frac{52!}{5! \times (47!)}$$

Key in: 52 n! ÷ (5 n! × 47 n! =

Answer: 2,598,960 hands

Of these poker hands, how many would contain 3 Kings? There are 3 out of 4 ways to choose the 3 Kings and 2 out of 48 ways to choose the remaining cards.

$$\frac{4!}{3!\ (4-3)!} \times \frac{48!}{2!\ (48-2)!} = \frac{4!}{3! \times 1!} \times \frac{48!}{2! \times 46!}$$

Note: $1! = 1$; $2! = 2 \times 1 = 2$; $3! = 3 \times 2 \times 1 = 6$

Key in: 4 n! ÷ 3 n! × (48 n! ÷ (2 × 46 n! =

Answer: 4512 hands

What is the probability of getting a hand with 3 of a kind? There are 13 different kinds of cards; the number with 3 of a kind would 13 × 4512. Divide by the total number of possible hands to get the probability.

Key in: 4512 × 13 ÷ 2598960 =

Answer: 0.022569027

Key in: 1/*x*

Answer: 1 in 44.3 chances (round to 45)

log ln Logarithm Keys

A number x can be expressed as a base b raised to a power y:

$$x = b^y$$

y is called the logarithm of x and is symbolized as:

$$y = \log_b x$$

The logarithm of a number is the power to which some base must be raised to equal that number. Logarithms and exponential functions have numerous uses in mathematics. Logarithms simplify arithmetic and algebraic computations according to the following rules:

$$\log_b (xy) = \log_b x + \log_b y$$

$$\log_b x/y = \log_b x - \log_b y$$

$$\log_b x^y = y \log_b x$$

The graphs of two logarithm functions appear in Figure 1.1.

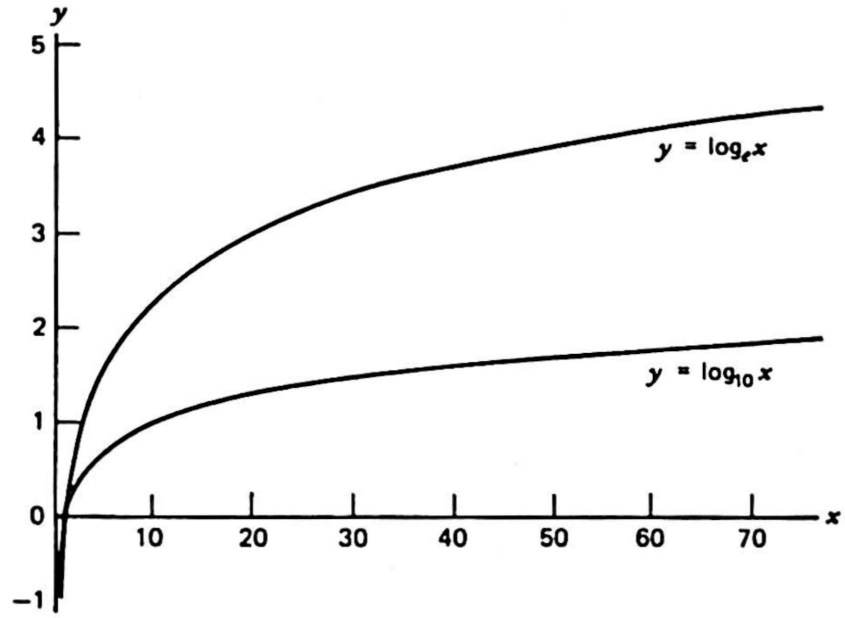


Figure 1.1 Logarithm functions

Using tables of logarithms multiplication, division, exponentiation, and root extraction can be accomplished by addition and subtraction. The logarithmic tables are built into the calculator for the two most common bases 10 and e. The number e is an irrational number resulting from an infinite series and it cannot be expressed in exact decimal or fractional form; it is approximately equal to 2.71828. $\text{Log}_e\, x$ is expressed as $\ln x$ or the natural logarithm. The base 10 logarithm is expressed as $\log x$ and is known as the common logarithm.

10^x Base 10 Inverse Logarithm or Antilogarithm

e^x Base *e* Inverse Logarithm or Antilogarithm

The inverse logarithm, or antilogarithm, is the number generated when the base is raised to the power of the logarithm. 10^x is the antilogarithm for $\log x$ and e^x is the antilogarithm for $\ln x$.

Calculate: 3×4 using log and ln.

Key in:	Answer:
3 **log**	0.477121254
+ 4 **log**	0.602059991
=	1.079181246
10^x	12.
3 **ln**	1.098612289
+ 4 **ln**	1.386294361
=	2.48490665
e^x	12

The **10^x** key can also be used to solve for powers of 10.

Calculate:	Key in:	Answer:
10^4	4 **10^x**	10000
10^0	0 **10^x**	1
e^0	0 **e^x**	1
e^1	1 **e^x**	2.718281828
10^{-1}	1 **+/−** **10^x**	0.1
10^{-5}	5 **+/−** **10^x**	0.00001

The logarithmic scale is often used when the range of values of a phenomenon is too large to be graphed meaningfully on an arithmetic scale. The decibel is a logarithmic scale unit that expresses the relative magnitude of two sounds. In the audio field, the decibel is used to describe the response of an amplifier in terms of the deviation of the output voltage over some frequency range from the output voltage at some standard frequency, using a constant amplitude input test signal.

$$\mathrm{dB} = 20 \log_{10} \frac{V_f}{V_s}$$

where V_f = voltage at a given frequency

V_s = voltage at a standard frequency

The dynamic range of an amplifier is 30Hz to 15,000Hz ± 0.3 dB. The power output at a standard frequency of 1,000 Hz is 8 watts with an 8-ohm load. The amplifier is tested at a frequency of 1,000 Hz with the same input voltage and the output voltage is found to be 7 volts. Does the test indicate that the manufacturer's specifications are being met?

Using the power formula:

$$P = \frac{V_s^2}{R}$$

$$8 = \frac{V_s^2}{8}$$

$$V_s = \sqrt{8 \times 8} = 8 \text{ volts}$$

Key in: 20 x (7 ÷ 8) log =

Answer: −1.159838939 dB

The manufacturer's specification is −0.3 dB. Therefore, the specifications are not being met. What minimum value would the test have to yield to meet the specifications?

$$V_f = \frac{\text{dB}}{10^{20}}$$

$$V_f = 8 \times 10^{\frac{-0.3}{20}}$$

Key in: 8 x (.3 +/− ÷ 20) 10^x =

Answer: 7.728407032 volts

sin cos tan Sine, Cosine, Tangent

The basic definitions of sine, cosine, and tangent are: sine x = length of opposite side/lengh of hypotenuse = O/H, cosine x = length of adjacent side/length of hypotenuse = A/H, tangent x = length of opp. side/length of adjacent side = O/A.

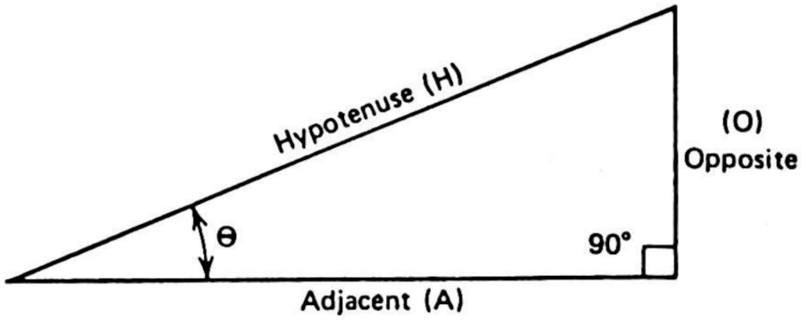


Figure 1.2 Right Triangle

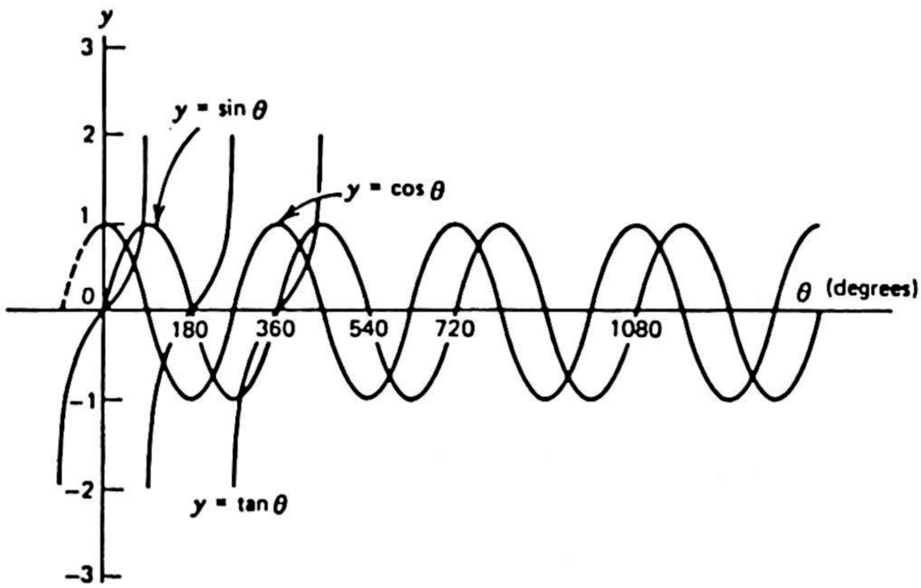


Figure 1.3 Sine, Cosine, Tangent

Figure 1.3 illustrates the graph of these functions.

Calculate: sine of 30°, cosine of 23°, tangent of 90°

Key in:	*Answer:*	**Mode:** DEG
30 **sin**	0.5	
23 **cos**	0.920504853	
90 **tan**	**E**	(Tangent of 90° is undefined and cannot be calculated.)

Angles are specified in degrees, radians, or Grad (popular in Europe).

Degree: $1°$ = 1/360 of a circle = $\pi/180$ radians = 0.017453292 radians

Radian: 1 rad = $1/2\pi$ of a circle = $180/\pi$ = 57.29577951 =

Grad: 1 grad = 1/400 of a circle

Key in: 57.29577951 **1/x** *Answer:* 0.017453292

The pilot of a small plane flying at a height of 1000 feet observes an airport at an angle of 5°. How far away is the airport?

Note: Angle A = 90 − 5 = 85°

tan 85 = a/1000

a = tan 85 × 1000

Key in: 85 **tan** **×** 1000 **=**

Answer: 11430.0523 feet

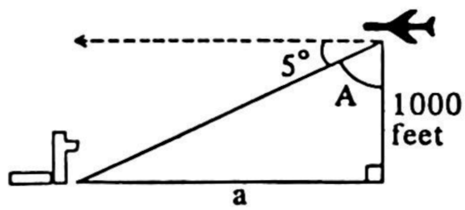


Figure 1.4 Plane

The angle of elevation of a ladder is 75° and it is placed 4 feet from the base of a building. How long is the ladder?

Note: cos 75 = 4/b

b = 4/cos 75

Key in: 4 **÷** 75 **cos** **=**

Answer: 15.45 feet

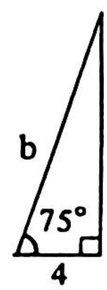


Figure 1.5 Ladder

The diagonal distance of a television screen is 19 inches. The angle subtended by the shorter side is $\pi/5$ radians. What is the length of each side?

$\sin \pi/5 = a/19$

$a^2 + b^2 = c^2$

$b = \sqrt{c^2 - a^2}$

Key in:

DRG to change to **RAD**

a. **(** **π** **÷** 5 **)** **sin** **×** 19 **M+**

b. **(** 19 x^2 **−** **RM** x^2 **)** $\sqrt{\ }$

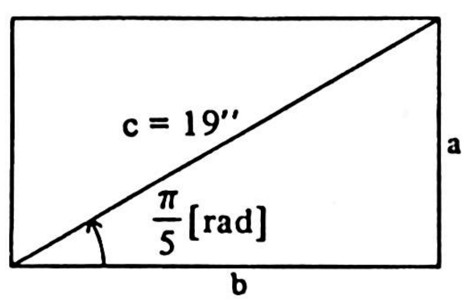


Figure 1.6 TV Screen

Answer: a = 11.168 inches

b = 15.371 inches

sin⁻¹ cos⁻¹ tan⁻¹ Inverse Trigonometric Functions

The inverse of sine, cosine and tangent are arcsine ($\sin^{-1}$), arccosine ($\cos^{-1}$) and arctangent ($\tan^{-1}$).

Calculate: $\sin^{-1}$ of 0.5, $\cos^{-1}$ of 0.5 and $\tan^{-1}$ of 1.

Key in:	*Answer:*
DRG to change to **DEG**	
.5 **$\sin^{-1}$**	30°
.5 **$\cos^{-1}$**	60°
1 **$\tan^{-1}$**	45°

Note: These are the "principal" values. The inverse trigonometric functions can represent many values, e.g., $\sin^{-1}$ .5 is 30°, 150°, 390°, 510° and also −210°. The lowest, or principal value of the inverse is always produced by the calculator. The user must decide which one of several values is the answer, or if several values are required.

Calculate: $\sin^{-1}$ (sin 60°)

Key in: 60 **sin** **$\sin^{-1}$**

Answer: 60

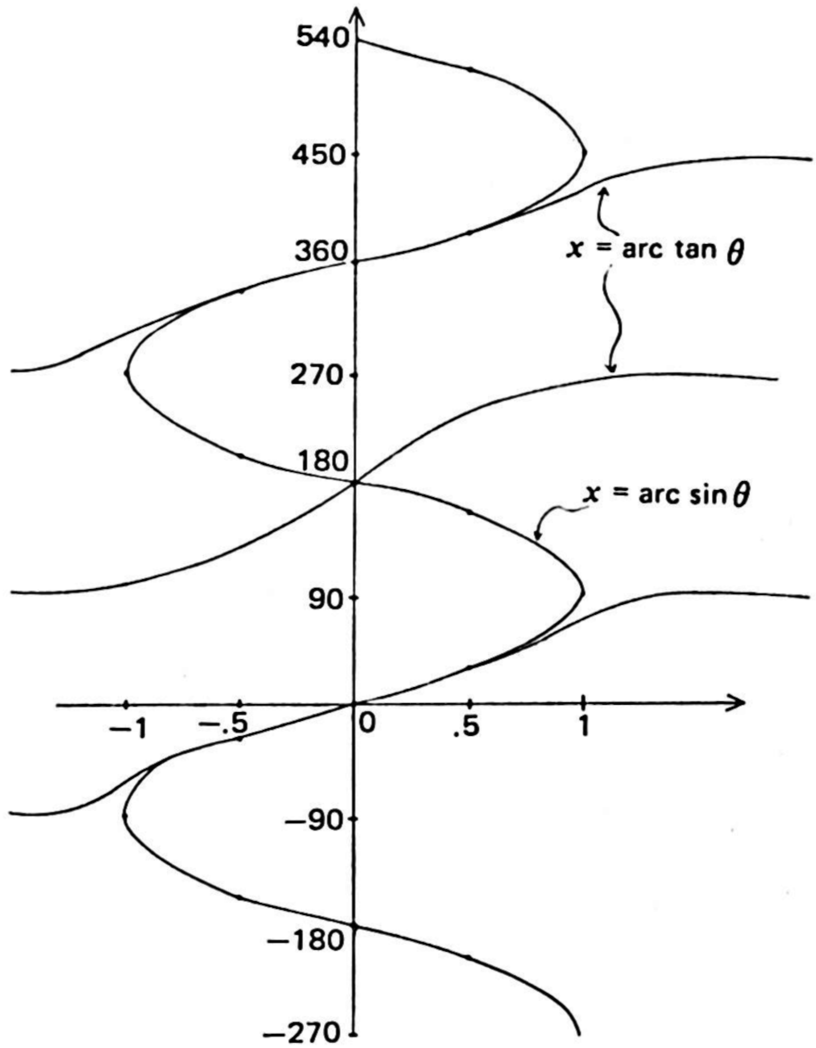


Figure 1.7 Inverse Trig. functions

The center of the sun is 92.8 million miles from earth. The sun has a diameter of 864,000 miles. What angle does the sun subtend from earth?

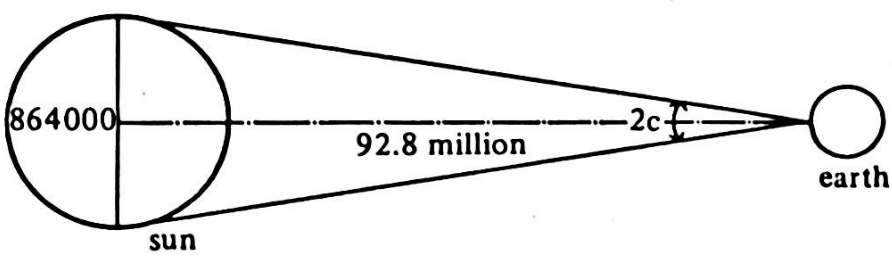


Figure 1.8 Sun-earth

$$\tan C = \frac{43.2 \times 10^4}{92.8 \times 10^6}$$

Key in: (43.2 EXP 4 ÷ 92.8 EXP 6) $\tan^{-1}$

Answer: 0.2667°

Key in: × 2 =

Answer: 0.5334°

The driveway of a house will start in a corner and curve around to the other corner. (See Figure 1.9) What is the outside length of the driveway?

The driveway is an arc (section of a circle). The distance (l) subtended by the arc is called a chord, and in this case is 200 feet. The rise (h) of the chord is:

h = 100 – 60 = 40 feet

s = length of the arc

R = radius of the full corcle

d = $R - h$ = $R - 40$

l = length of chord = $2\sqrt{R^2 - d^2}$ = 200 feet

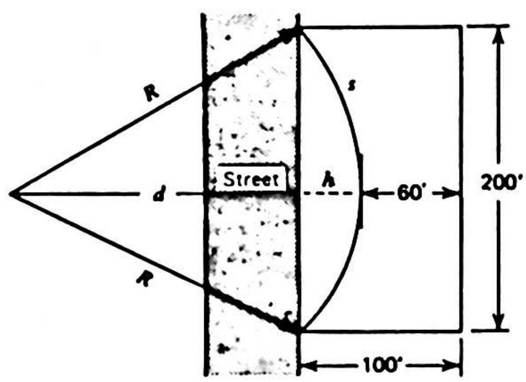


Figure 1.9 A driveway which is a chord of a circle.

Solve for R:

$$200 = 2\sqrt{R^2 - d^2} \qquad 100 = \sqrt{R^2 - d^2}$$

$$10000 = R^2 - d^2 = R^2 - (R - 40)^2 =$$

$$R^2 - R^2 + 80R - 1600$$

$$80R = 11600$$

$$R = 11600 \div 80 = 145 \text{ feet}$$

The length of the arc of a circle is given as:

$$s = 2R \cos^{-1} \frac{d}{R}$$

Solve for s:

Mode: RAD

Key in: $2 \times \overline{145} \times ((\overline{145 - 40}) \div 145) \cos^{-1} =$

(R = 145; d = 145 − 40)

Answer: 220.69 feet (length of arc).

[π] Pi

The value of π can be approximated as 3.141592654 which is close enough for most purposes. (Sometimes a less accurate approximation, 22/7, is used.) The π value is included on scientific calculators because of its wide use in geometry and trigonometry. It is defined as the ratio of the circumference (c) of a circle to its diameter (D) and holds true for all size circles.

$$C = \pi D = 2\pi R \,, \quad \text{where} \quad R = \text{radius}$$

Calculate the circumference of a circle with a radius of 6.

Key in: 2 × π × 6 =

Answer: 37.69911184

[→DEG] [→D.MS] Time Conversions

Angles that are not whole numbers are usually given in terms of degrees, minutes and seconds. There are sixty minutes in one degree. Angles given in terms of minutes must be converted to a decimal equivalent before the trigonometric functions can be used.

When the calculator is used for converting, the first two places after the decimal point represent minutes and the next two represent seconds.

Convert 35°5′26″ to its decimal equivalent.

Key in: 36. 05 (*minutes*) 26 (*seconds*) →DEG

Answer: 36.09055556

The reverse is to convert to degrees, minutes and seconds.

Key in: 36.09055556 →D.MS

Answer: 36. (*degrees*) 05 (*minutes*) 26 (*seconds*) (or 36°5′26″)

Two boats (see Figure 1.10) are observed from a tower 75 meters above a lake. The angles of depression observed from the vertical are 12°30′ and 7°10′. How far apart are the boats? Find the horizontal distance of each boat from the tower. The difference will be the distance they are apart.

First boat: $\tan 7°10' = \dfrac{z}{75}$

Solve for z:

Key in: 7.1 →DEG tan x 75 =

x→M

Answer: 9.430377057

Second boat: $\tan 12°30' = \dfrac{y}{75}$

Solve for y:

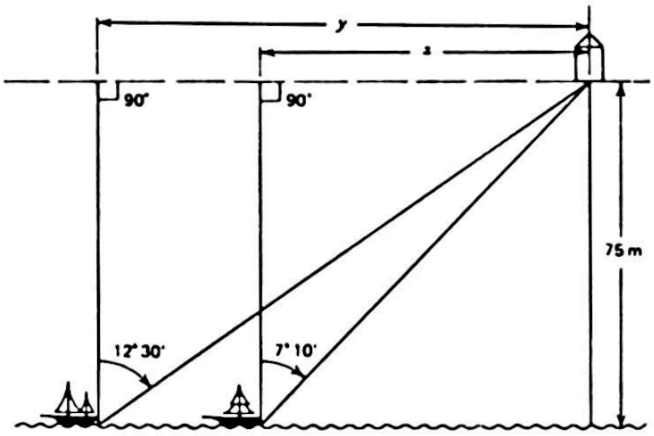


Figure 1.10 Two boats

Key in: 12.3 →DEG tan x 75 =

Answer: 16.6270997

Key in: – RM =

Answer: 7.196722643 meters apart

These keys are also used when time needs to be accumulated or averaged. Hours can take the place of degrees in "DMS".

A track team member is training to run the marathon. Time trials this year have produced the following times: 3 hours 42 minutes 6 seconds, 3 hours 15 minutes 25 seconds, 3 hours 5 minutes 48 seconds, and 2 hours 59 minutes 42 seconds. What is the runner's average time?

Key in:			Display:	
	3.4206	→DEG	3.701666667	decimal equivalent
+	3.1525	→DEG	3.256944444	decimal equivalent
+	3.0548	→DEG	3.096666667	decimal equivalent
+	2.5942	→DEG	2.995	decimal equivalent
=			13.05027778	hours total
÷ 4 =			3.262569445	hours
→D.MS			3. 15 45 25	
			which is 3 hours 15 minutes 45 seconds	

The triangle which we have been using has been a special one because it contains a right angle. A more general triangle is shown in Figure 1.11. Two powerful theorems are applicable, called the "law of sines" and the "low of cosines".

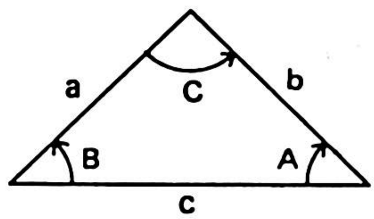


Figure 1.11
General triangle

Two wires are run from the side of a building to the street below. One wire is 42 feet long and makes an angle of 58= with the horizontal. The other wire is 48 feet long. What is the distance between the anchor points and the angle of the second wire with the horizontal?

B = 180° − 58° = 122°

$$\frac{48}{\sin 122^\circ} = \frac{42}{\sin C}$$

sin C = 42/48 x sin 122°

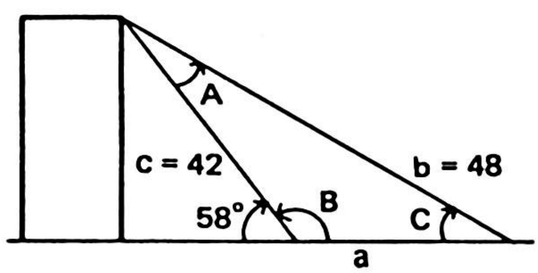


Figure 1.12 Two Wires

Mode: **DEG**

Key in:

C: 42 ÷ 48 x 122 **sin** = **sin⁻¹** **x→M**

Answer: 47.9

A: 180 − **RM** − 122 = **x→M**

Answer: 10.09°

a: 48 ÷ 122 sin x RM sin =

Answer: 9.920 feet

Note: In the above example, an angle greater than 90° occurs, placing it the second quadrant. Quadrants are created when a circle is placed over rectangular coordinates (see Figure 1.13). The four equal parts are called "quadrants". The line representing the positive part of the x-axis as it travels in a counter-clockwise direction is the measurement of an angle.

The quadrant specifies the sign for each of the trigonometric functions as follows:

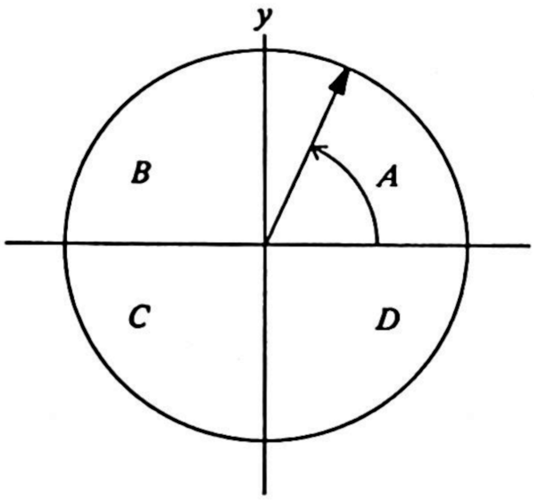


Figure 1.13 Quadrans

Quadrant	angle range (°)	sine	cosine	tangent
A	0 to 90	+	+	+
B	90 to 180	+	–	–
C	180 to 270	–	–	+
D	270 to 360	–	+	–

The scientific calculator automatically gives the correct sign.

Prove the following identity for A = 220° and B = 100°.

$$\tan\ (A - B) = \frac{\tan A - \tan B}{1 + \tan A \tan B}$$

Key in:

a. (220 − 100) tan

Answer: −1.732050808

b. C.CE (220 tan − 100 tan) ÷ (1 + 220 tan x 100 tan =

Answer: −1.732050808

A yacht sails out to sea for twenty miles at the beginning of the day. On the return trip a strong current pushes the yacht 10° off course. After sailing for 19 miles on the return trip, how far is the yacht from its starting point?

Solve for c using the Law of Cosines:

Key in:

$$\overset{a}{20}\ x^2 + \overset{b}{19}\ x^2 - 2 \times \overset{a}{20} \times \overset{b}{19} \times \overset{c}{10}\ \cos = \sqrt{\ }$$

Answer: 3.542 miles from the original point

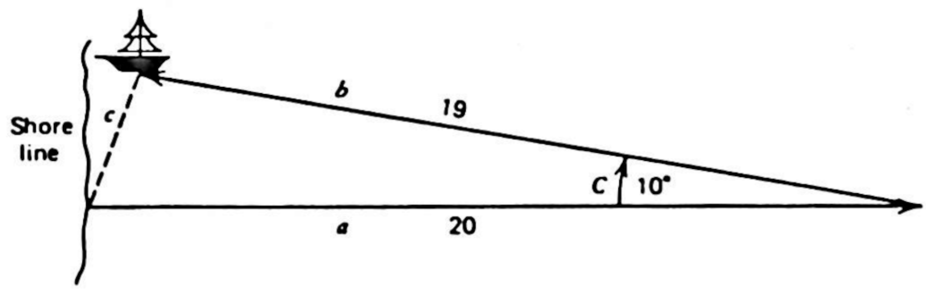


Figure 1.14 Triangle-Law of Cosines

[→rθ] [→xy] Polar Coordinate – Rectangular Coordinate Conversion Keys

The most common coordinate system is the cartesian or rectangular system. The location of a point is given by its distance from each axis. The distance from the y-axis (abscissa) is given first followed by the distance from the x-axis (ordinate). The coordinates of the point in Figure 1.15 are (2, 3).

The system known as polar coordinates identifies a location by using an angle and a distance. (See Figure 1.16).

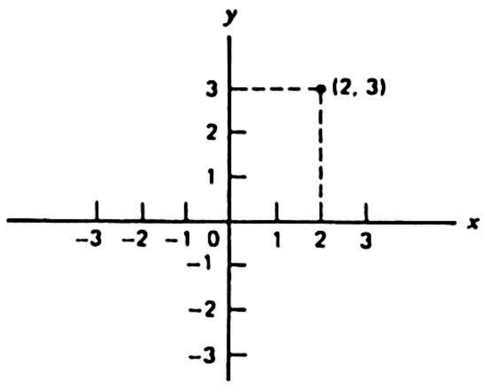


Figure 1.15 Rectangular coordinate system

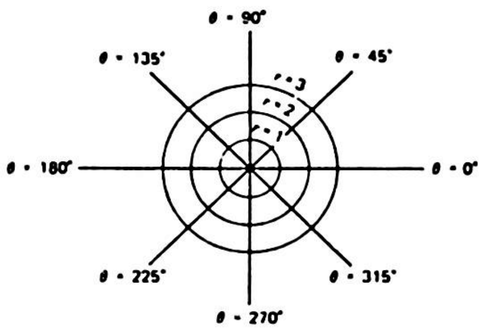
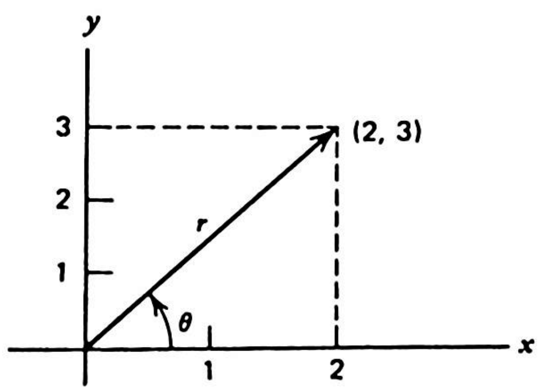


Figure 1.16 Polar coordinate system

The polar coordinates can be superimposed on the cartesian plane as in Figure 1.17.

$$(2, 3) = (r, \theta)$$



By using trigonometry, the conversions from one system to another can be developed. To convert (2, 3) into polar coordinates:

Mode: DEG

Key in: 2 ↕ 3 →rθ

Answer: r = 3.605551275

To get the angle: ↕ $\theta = 56.30993247°$

To do the inverse, changing polar coordinates to rectangular:

Mode: DEG

Key in: 3.6 ↕ 56.3 →xy

Answer: $x = 1.99744$, which is approximately 2

To get the y coordinate: ↕ $y = 2.995$, which is approximately 3.

The coordinates of a new position are obtained by using the polar rectangular conversion keys.

In Figure 1.18 the original coordinates $(x, y) = (1.5, 2)$, the angle of the new position is $\theta = 60°20'14''$, while $l = 7$. What are the coordinates of (x_i, y_i)?

Key in: 7 ↕ 60.2014 →DEG →xy

Display: 3.4642598

Key in: x→M ↕

Display: 6.082672442

Key in: + $\overset{y}{2}$ =

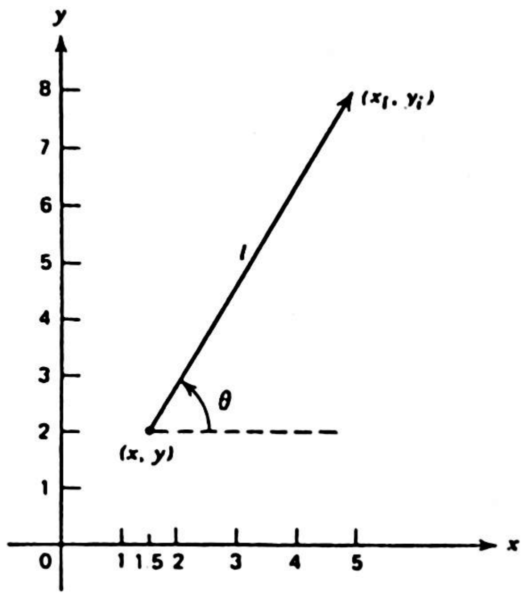


Figure 1.18 Coordinates-new position

Display: 8.082672442

Key in: RM + $\overset{x}{1}$.5 =

Display: 4.9642598

Answer: (x_i, y_i) = (4.964, 8.083)

The dot product of a pair of vectors yields a scalar quantity.

$$V_1 \cdot V_2 = C$$

where $C = |V_1| \times |V_2| \times \cos\theta$

where $\theta = \measuredangle V_1 - \measuredangle V_2$

The coordinates of V_1 are (0, 0), (3, 5) and for V_2 they are (0, 0), (2, 1). What is the resultant scalar quantity?

Key in:	*Answers:*
3 ↕ 5 →$r\theta$	r = 5.83
↕	θ = 59.04°
2 ↕ 1 →$r\theta$	r = 2.24
↕	θ = 26.57°
5.83 x 2.24 x (59.04 −	
26.57) cos =	11.02 (scalar quantity)

A merchant ship steams out of port on a course of 25° west of north. After 35 miles course is changed to 5° east of north and 300 miles are covered. What is the total distance traveled?

Note: In navigation, the course is specified with north and south directions measured clockwise from the north axis. Counter-clockwise angles are negative.

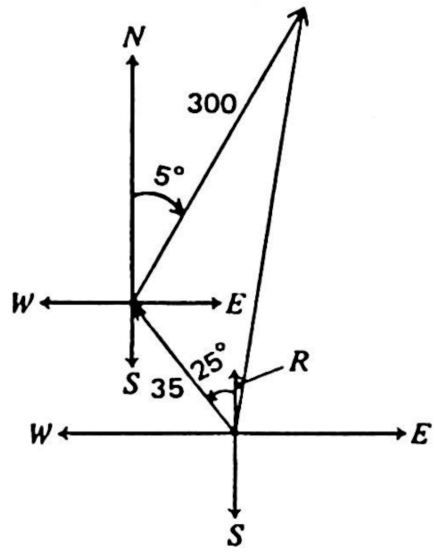


Figure 1.19 Ship's Route

Key in:		X	Y
35 ↕ 25 +/– →*xy*		31.72	
↕			–14.79
300 ↕ 5 →*xy*		298.86	
↕			26.15
	add:	330.58	11.36
330.58 ↕ 11.36 →$r\theta$		330.775 miles from the starting point	
↕		1.968° course (r)	

Hyperbolic Functions

[hyp] [sin], [hyp] [cos], [hyp] [tan] **Sinh, Cosh, Tanh**

Certain curves found in nature are called "catenary." The catenary and its shape are particular combinations of exponential functions found so often in engineering and physics that a special name, *hyperbolic*, has been given to these curves.

Hyperbolic sine of x = $\sinh x$ = $\dfrac{1}{2}(e^x - e^{-x})$

Hyperbolic cosine of x = $\cosh x$ = $\dfrac{1}{2}(e^x + e^{-x})$

Hyperbolic tangent of x = $\tanh x$ = $\dfrac{e^x - e^{-x}}{e^x + e^{-x}}$

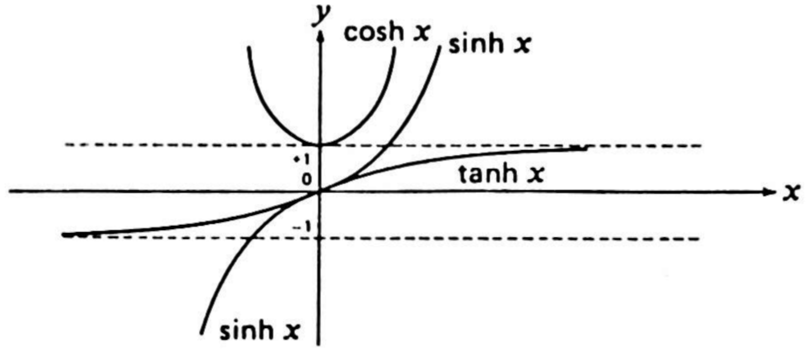


Figure 1.20 Hyperbolic functions

Figure 1.20 illustrates the graph of these functions. Calculate sinh 6 and prove the relationship to e^x.

Key in: 6 **hyp** **sin**

Answer: 201.7131574

Key in: .5 × (6 e^x − 6 +/− e^x =

Answer: 201.7131574

The inverse hyperbolic functions are written: $\sinh x^{-1}$, $\cosh x^{-1}$, $\tanh^{-1} x$.

Calculate: $\sinh^{-1} x$ if $x = 201.7131574$.

Key in: 201.7131574 **archyp** **sin**

Answer: 6

→HEX →DEC A, B, C, D, E, F Hexadecimal Conversion

Pressing the **→HEX** key will place the calculator into the hexadecimal-base-system mode: Anytime the calculator is in this mode **HEX** will appear at the right

of the system. Changing to the **HEX** mode with a number in the display will automatically convert that number to its hexadecimal equivalent. Leaving the hexadecimal mode by pressing **→DEC** will change any number in the display to the decimal equivalent.

The need for other numbering systems than decimal (base 10) has been primarily brought about by the ever growing development and use of computers. The smallest basic component of a computer is the bit. Each bit is either on or off, or in numerical terms, a value of 0 or 1 (the binary number system, base 2). For convenience and efficiency most computers are composed of bytes which are bits in multiples of two. With few exceptions, bytes are either 8-bits or 16-bits. Programming a computer in machine language requires the hexadecimal numbering system.

In every numbering system, each position in a number has a value. For instance in the decimal system, (starting for the right) the first digit is multiplied by 1; the next by 10; the third by 100 and so forth. In the hexadecimal system, the values are 1, 16, 256, 4096, etc.

The equivalent numbers for each of four base systems up to decimal value 20, are shown in the table. Note the use of letters A to F in hexadecimal.

Convert ABCDEF to decimal:

Key in:	**Display:**
→HEX A B C D E F →DEC	11259375.
→HEX	ABCDEF. HEX

Mutiply the hexadecimal numbers: CAD × BED × 125

Key in:

→HEX CAD × BED × 125 **=**

Answer: AD0465ED

Positive hexadecimal values on the EL-5500II are limited to between 0 and 2540BE3FF (9999999999.). In a computational machine based on the charged or

uncharged bit, there is no such condition as "negative".

Consequently, the highest possible hexadecimal value (FFFFFFFFFF) is −1. Working in the direction of lower hexadecimal values, the limit is FDABF41C01, which is the decimal value limit of −9999999999.

Key in:	Display:
1 +/− →HEX	FFFFFFFFFF
→DEC 2 +/− →HEX	FFFFFFFFFE
→DEC 9999999999 +/− →HEX	FDABF41C01

Table of Values for Four Base Systems

Decimal	Binary	Octal	Hexadecimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14

STAT Statistical Mode

DATA CD (x, y) Σy Σy^2 Σx Σx^2 n Σxy $\bar{y}$ $\bar{x}$ Sy Sx σy σx r a b x' y'

Twenty Statistical Keys

IMPORTANT INFORMATION: HAND CALCULATIONS IN STAT MODE. The EL-5500II allows use of both MEAN and STANDARD DEVIATION in hand calculations. Any calculations involving other statistical functions must be performed in one of two ways:

1) write down the values and then key them in by hand.

2) switch to RUN and calculate using:

U	Σy^2	W	Σxy	Y	Σx
V	Σy	X	Σx^2	Z	n

The calculator mode on the EL-5500II has the capability of both single-variable and two-variable statistics.

Single-variable Statistics

$\bar{x}$ represents the mean, or arithmetic average. The formula for $\bar{x}$ is:

$$\bar{x} = \frac{x_1 + x_2 + x_3 + \cdots x_n}{n}$$

n is the number of samples. If the data base is small, it could be the total number. x_1 is the first data point, x_2 is the second up to x_n, which is the nth or last data point. The symbol Σ (sigma) means "sum of" and when Σx is pressed, the total of all the data is retrieved.

The statistical techniques available in the calculator mode are based on the concept of central tendency. A large enough data base will result in a frequency of data about the mean with values concentrated close to the mean, resulting in the familiar normal (or bell-shaped) curve illustrated in Figure 1.21.

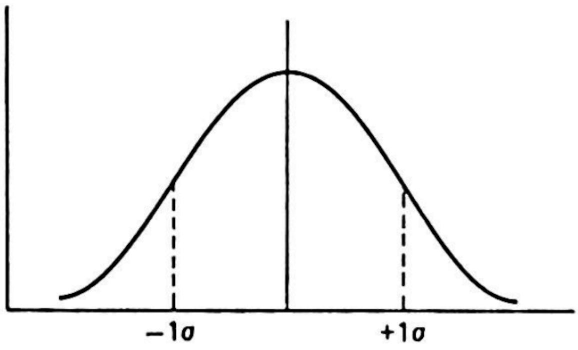


Figure 1.21 Normal Distribution

The standard deviation is the measure of the deviance of the data from the mean. A large deviance implies a group with less cohesion than the case where the deviance is small. One standard deviation (plus and minus) includes 68.27% of the area under the normal curve. Two standard deviations will include 95.45% of the total and 3 includes 99.73%. **Sx** is the standard deviation of the sample and **σx** the standard deviation of the population. Logically, the standard deviation of the whole population will always be less than the standard deviation of a sample. The basic formulas for standard deviation are:

$$\sigma = \sqrt{\frac{\Sigma x^2 - n\bar{x}^2}{n}} \qquad s = \sqrt{\frac{\Sigma x^2 - n\bar{x}^2}{n-1}}$$

These calculations can not be performed by hand in STAT mode (see IMPORTANT INFORMATION). The general procedure for obtaining the average and standard deviation for a set of data is:

1. Place the calculator in the **STAT** mode.
2. Place the data into the calculator.
3. Press $\bar{x}$ for the average and **Sx** for standard deviation.

The price of an eight ounce jar of instant coffee was recorded each month of the year. Calculate the average price and the standard deviation from the prices: \$3.65, \$4.20, \$4.20, \$4.85, \$3.90, \$3.90, \$2.95, \$2.80, \$3.90, \$3.50, \$3.50, \$3.20.

Mode: **STAT**

Key in: 3.65 **DATA** 4.2 **DATA** 4.2 **DATA** 4.85 **DATA** 3.9 **DATA**···
(Continue to place in above data)

Answer: $\bar{x}$ = 3.7125 = average

σx = 0.551182441 = standard deviation for the population since the sample is the entire population.

Under the assumption that the sample is large enough for the normal curve to apply, ranges for the prices can be obtained by taking the mean and then adding or subtracting the standard deviation.

Key in:	*Answer:*
$\bar{x}$ − σx =	3.1613
$\bar{x}$ + σx =	4.2637

Approximately 68% of the prices during the year fell between \$3.16 and \$4.26. How many standard deviations will encompass all of the sample prices? Subtract the lowest data point from the average and divide by the standard deviation for the negative number. For the positive number, take the highest data point, subtract the average and divide by the standard deviation. If the value obtained is fractional, always round up. For example, if it is 1.2, the number of standard deviations is 2.

Key in:	*Answer:*
$\bar{x}$ − 2.8 = ÷ σx =	1.6555 (or −2 standard deviations)
4.85 − $\bar{x}$ = ÷ σx =	2.0637 (+3 standard deviations)

Data input with "weighting" is possible. Weighting is particularly valuable where a large number of samples have been collected of the same value. The value inputted is being weighted by entry more than once in a single operation.

Value	Number of Occurences
6	10
7	48
8	49
9	32
10	17

The number of times (N) the data occurs is entered as × N. After pressing **DATA** each time, the total number of data points entered will be displayed.

Key in:

STAT Mode

6 x 10 **DATA** 7 x 48 **DATA** 8 x 49 **DATA** 9 x 32 **DATA** 10 x 17 **DATA**

n is 156 (use **TAB** 3)

$\bar{x}$ is 7.987 mean

S_x is 1.101 standard deviation for the sample because the sample is a small part of a large population.

Standard deviation in this example means 68.27% of the values fall between:

Key in:		*Answer:*
	$\bar{x}$ + S_x =	9.088
		and
	$\bar{x}$ − S_x =	6.886
95.4% between	$\bar{x}$ + 2 x S_x =	10.190
		and
	$\bar{x}$ − 2 x S_x =	5.785
99.73% between	$\bar{x}$ + 3 x S_x =	11.291
		and
	$\bar{x}$ − 3 x S_x =	4.683

CD means "correct data." If the last data entry was in error or it will be changed, press **CD**. The **CD** key may be used as **DATA** entries are being made or directly after a statistical function. The number of samples will be shown, less one. Now proceed as before to enter new data. If weighting has been used, the entire original entry should be keyed in; for example, if 10 x 17 has been keyed in and **CD** is pressed, all seventeen of these entries will be eliminated.

Two Variable Statistics:

Two-variable statistics makes possible the development of a relationship (correlation) between two sets of data. Each pair of data has an x and y value.

From these sets of data a line of regression can be established. The relationship of the two sets of data by use of the strait line method is called "Linear Regression". The equation of the strait line is $y = a + bx$, where a is the point at which the line crosses the Y-axis and b is the slope of the line.

The correlation coefficient is called "r".

$$r = \frac{n\,\Sigma(xy) - \Sigma(x)\,\Sigma(y)}{\sqrt{[n\Sigma x^2 - (\Sigma x)^2]\,[n\Sigma y^2 - (\Sigma y)^2]}}$$

A perfect correlation between two values in linear regression results in an r equal to 1 (−1 is a perfect negative correlation); in other words, by knowing the value of one variable you can predict with 100% accuracy the value of the other variable. The further the value of r is from 1, the less reliable will your predictions be. The following table can be used as a set of definitions of the values of the correlation coefficient:

	Value of r	**Call it**
Positive Correlation	+0.80 to 1.00	Extra High
	+0.60 to 0.80	High
	+0.40 to 0.60	Moderate
	+0.20 to 0.40	Low
	−0.20 to +0.20	Nil
Negative Correlation	−0.20 to −0.40	Low
	−0.40 to −0.60	Moderate
	−0.60 to −0.80	High
	−0.80 to −1.00	Extra High

Is weight a good predictor of longevity among men 65 years of age? In 1950, 10 men, each six feet tall, were selected for an experiment to determine if their weight effected their life span.

Sample	1	2	3	4	5	6	7	8	9	10
Age at death	72	67	69	85	91	68	77	74	70	82
Weight at age 65	185	226	200	169	170	195	175	174	198	172

Key in: STAT Mode

72 (x, y) 185 DATA

67 (x, y) 226 DATA

etc.

r −0.793

r indicates a relatively high negative correlation. Higher weight indicates a shorter life span. To graph the regression line, a and b are used.

Key in:	*Answer:*
a	321.929 (y-axis)
b	−1.795 (slope)

Predict the age of death of a 6-foot man weighing 190 pounds in 1950.

Key in:	*Answer:*
190 x'	73.495

To reach age 90, what should a man's weight be in 1960?

Key in:	*Answer:*
90 y'	160.371 pounds

To reach age 150, what should a man's weight be? Obviously, the answer will make no sense, indicating the danger of carrying a straight-line extrapolation too far.

A chemist measures the pressure in a gas filled container as it is heated. From the ideal gas equations, he knows that as temperature increases and volume is held

constant, pressure will also increase. Calculate the average increase in pressure (Y) per degree Celsius increase in temeperature (X).

The slope of the regression line will be the increase.

X = Temperature °C	23	28	35	37	58	78	100
Y = Pressure Kg/cm^2	75	79	81	81	87	92	98

Key in: **STAT MODE** 23 (x, y) 75 **DATA**
28 (x, y) 79 **DATA**
: :
100 (x, y) 98 **DATA**

n = 7 data entries
slope = **b** = 0.279 $Kg/cm^2/°C$
coefficient = **r** = 0.994 indicates excellent data fit to least squares line.

Exponential Regression

Data points having a linear relationship can be fitted to a regression line. Many situations arise where the data can be fitted more closely to the exponential curve $y = a e^{bx}$.

Exponential data is entered by taking the natural log of the y value and the equation becomes:

$$\ln y = \ln a + bx$$

Estimated results are obtained by raising the result exponentially. The value in register a will be $\ln a$.

Determine the actual equation, coefficient of correlation, and estimated y values for $x = 9.2$ and -2.6, for the following values:

x	0.5	1.6	3.6	7.9	8.7
y	8	13.2	52.9	1008	2201

Key in: STAT

.5	(x, y)	8	ln	DATA
1.6	(x, y)	13.2	ln	DATA
3.6	(x, y)	52.9	ln	DATA
7.9	(x, y)	1008	ln	DATA
8.7	(x, y)	2201	ln	DATA

Key in:	*Answer:*
r	0.999
a	1.583
b	0.687
a e^x	4.868

$$\ln y = \ln a + bx = 1.583 + 0.687x$$

$$y = 4.868\, e^{0.687x}$$

Obtain estimated values of y:

Key in:	Estimated y:
9.2 y' e^x	2713.165
2.6 +/– y' e^x	0.815

PART II

BASIC Mode

Calculating can be performed in the **RUN** Mode of BASIC without programming. The technique of placing a formula into the computer under **RUN** is the same format as in **LET** or **PRINT** commands in programming.

Many of the numeric calculation keys have abbreviations peculiar to BASIC and will appear on the display in that manner.

Function	Key	BASIC Term
Inverse sine	$\sin^{-1}$	ASN
Inverse cosine	$\cos^{-1}$	ACS
Inverse tangent	$\tan^{-1}$	ATN
Hyperbolic sine	hyp sin	HSN
Hyperbolic cosine	hyp cos	HCS
Hyperbolic tangent	hyp tan	HTN
Inverse hyp sin	archyp sin	AHS
Inverse hyp cos	archyp cos	AHC
Inverse hyp tan	archyp cos	AHT
Reciprocal	$1/x$	RCP
Antilogarithm-base e	e^x	EXP
Antilogarithm-base 10	10^x	TEN
Exponential	EXP	E
Polar Coordinates	$\rightarrow r\theta$	POL
Rectangular Coordinates	$\rightarrow xy$	REC
Factorial	n!	FACT
Power	y^x	^
Roots	$\sqrt[x]{y}$	ROT
Square root	$\sqrt{\ }$	SQR
Cube Root	$\sqrt[3]{\ }$	CUR
Squared	x^2	SQU

The sides of a triangle are 10 and 15 and the angle between them is 75°. Find the third side using the Law of Cosines.

$$c = \sqrt{a^2 + b^2 - 2ab \cos C}$$

Key in: A = 10 , B = 15 , C = 75 , √ (SQU A + SQU B − 2 * A * B * COS C) ENTER

Answer: 15.7275

The statistical and percent keys of the calculator will not operate in BASIC.

The 26 letter memories available in BASIC provide more variables than may be required by most programmers. Certain letters may be set aside for frequently-used constants. All values are retained even when the EL-5500II is turned off.

Place the metric conversion factors for miles to kilometers and inches to centimeters in memories M and N and convert 100 kilometers to miles and 50 inches to centimeters.

Key in:	M = 1.6093472 ,	miles to kilometers
	N = 2.54000508 ENTER	inches to centimeters

Note: The reverse conversions are accomplished by using the inverse (dividing by the constant).

Key in: 100 / M ENTER

Answer: 62.137 miles

Key in: 50 x N ENTER

Answer: 127.000 centimeters

CHAPTER 3

PROGRAMMING

Starting to Program — Modes, Line Numbers

An instruction line in a BASIC program is called a line of code. Each line of code begins with an identifying number. Usually these numbers are selected in multiples of 10. Sufficient room is thereby left to insert additional lines if this should become necessary.

ENTER (ENTER), END

To enter a simple program, e.g., assigning a value to a letter variable, key in in PRO mode:

10 Z = SIN 30 * 60 **ENTER**

The **ENTER** key is used in the programming mode to enter each line of code into the program. After **ENTER** is pressed a colon appears between the line number and the expression. The colon should never be pressed after a line number as it will always be inserted automatically by the computer.

The computer needs to know that a program has ended. To end the above program add the line:

20 **END ENTER**

LET

The expression for Z above is a **LET** statement. **LET** statements apply to both numeric and alphanumeric variables. **LET** is omitted except in a **THEN** statement as illustrated later. A series of **LET** statements on the same line are separated by colons.

RUN, BRK, STOP, CONT

The value of Z is calculated by changing to the **RUN** mode. Enter **RUN** and press **ENTER**. To display the value of Z, press Z and **ENTER**. The answer is 30.

BRK is the Break key. It is not programmable. Use it to stop a program in execution. The display will show:

"Break in (line number)"; indicating where the program has stopped.

STOP makes a program stop executing each time this command is encountered. **CONT** is used to restart the program execution.

PRINT, PAUSE, WAIT

The most common means of outputting data on the display is the command **PRINT. PRINT** is always followed by an expression. Without an expression an error will result.

The **PRINT** expression may be numeric or alphanumeric (a string). Quotation marks must be used with a string. Although an expression can be longer than 16 characters, only 16 will appear on the display.

A numeric expression may be a formula. A series of formulas can be included in a statement if separated by a comma or semicolons. The display automatically divides into two sections of 8 characters each when a comma is used. Therefore only two items can be displayed simultaneously with a comma separation. The semicolon must be used if more than two items are displayed at the same time. With semicolons, answers will appear on the display without separation. In the programming mode, key in:

```
10 X=5 : Y=2
20 PRINT 2*X, 5*Y
30 END
```

Change to **RUN** mode; press **RUN** and **ENTER**.

Answers:

```
          DEG
   10.          10.
```

Change Line 20 (PRO mode) and return (RUN mode).

```
20 PRINT 2 * X/4; 5 * Y; X/Y; -X * Y
```

Answers:

```
DEG
2.510.2.5-10.
```

The formula for the monthly equalized payment of a loan mortgage is:

Payment $= PV \div [1 - (1+i)^{-n}/i]$

where: PV = Present Value (amount of the loan)

i = Annual Interest Rate (decimal)

n = Number of Payments

To change to a monthly basis *i* is divided by 12 and *n* multiplied by 12.

What is the monthly payment on a $70,000 mortgage at 13% for 30 years? Using the BASIC commands we have learned so far, the program for this problem is:

```
10 PRINT "MORTGAGE PAYMENT"
20 V = 70000: I = .13: N = 30
30 P = V/((1 - ((1 + I/12) ^ (-12 * N)))/(I/12))
40 PRINT "PAYMENT $"; P
50 END
```

Change to the **RUN** mode; press **RUN** and **ENTER**. The display shows MORTGAGE PAYMENT. Press **ENTER** a second time to get the second **PRINT** statement which is the answer:

```
DEG
PAYMENT $774.339
```

PAUSE may be used instead of **PRINT**. **PAUSE** is a valuable command where it is desirable to make a statement and then have the program continue without having to press **ENTER**. Substitute **PAUSE** in Line 10 above and rerun the program.

The **WAIT** statement with a number permits the display of the **PRINT** statement for a specified period of times. 640 is the equivalent of 10 seconds. Insert **WAIT** 320: on Line 40 before **PRINT** and rerun the program.

LIST, Editing

LIST in the programming mode will produce the first line of your program. The same result is obtained by pressing the ↓ key, which will also take you through your program one line at a time. If you start with the ↑ key, the last line appears first. The advantage of the **LIST** command is it may be used with a line number to quickly access a particular statement in a large program.

The display will only show the first 16 characters of the line. To access the rest of the line the cursor keys are used. The cursor keys ◄(means ←) ►(means →) allow editing.

Change the values on Line 20 of the program to $7,000, .113, and 29. Go to Line 20 of the program. Press the right cursor key until the cursor is over a 0 in 70,000; **DEL** (delete). Continue to move the cursor to over the 3 in .13; **INS** (insert). To insert always move the cursor over the character that will follow the inserted character. A flashing top and bottom bracket appears; press 1. This value now is .113. Move the cursor over the 3 in 30; press 29 and **ENTER**. Rerun the program. The answer is $68.5435.

INPUT

In the mortgage payment example, the method used to place the data into the formula was limited. Variables may be used instead with the **INPUT** command. Change Line20 to: INPUT V, I, N and the program becomes:

```
10 PAUSE "MORTGAGE PAYMENT"
20 INPUT V, I, N
30 P = V/((1 - ((1 + I/12)^(-12 * N)))/(I/12))
40 WAIT 320: PRINT "PAYMENT $"; P
50 END
```

Now as the program is run, the display shows a ?. Place the value (70000) in and press **ENTER**. Continue for the next two values. Repeat changing only the interest rate to .17 (17%). The new answer is: $997.972.

Rather than have only question marks appear it is safer to name the variables.

To achieve this the format of Line 20 is changed to:

```
20 INPUT "V=";V, "I=";I, "N=";N
```

Run the program again — note the placement of semicolons and commas in Line 20.

The **INPUT** command also applies to strings. In the example include the owner's first name. Add:

```
 5 INPUT "NAME ?"; A$
15 PAUSE A$; "S HOUSE"
```

Return the program using your first name as the first input. To eliminate these or any lines, simply key in the line number (in PRO mode) and press **ENTER**, e, g, 5 **ENTER** 15 **ENTER**.

IF . . . THEN, GOTO, BEEP

The command **IF...THEN** allows decisions to be incorporated into a program. **IF** is followed by an equation or inequality and **THEN** by a statement number. Good programming technique dictates that the number of the statement following **THEN** should not be the number of the next statement in the program. In an **IF... THEN** statement the computer will test the condition stated and:

1. If true, will branch to the statement after **THEN**
2. If false, ignores the **THEN** clause and continues with the next line of the program.

The conditionals useful after **IF** in BASIC are:

Symbol	Meaning
=	Equals
<	Less than
>	Greater than
<>	Less or greater than (not equal to)
<=	Less than or equal to
>=	More than or equal to

Returning to the mortgage payment problem, to repeat the program without having to press **RUN** each time, add:

```
50 INPUT "REDO (Y/N) ?"; A$
60 IF A$="Y" THEN 20
```

The entire program now is:

```
10 PAUSE "MORTGAGE PAYMENT"
20 INPUT "V=";V, "I=";I, "N=";N
30 P=V/((1-((1+I/12)^(-12*N)))/(I/12))
40 WAIT 320: PRINT "PAYMENT $"; P
50 INPUT "REDO (Y/N) ?"; A$
60 IF A$="Y" THEN 20
70 END
```

Calculate the monthly payment for mortgages of $40,000; $50,000; and $60,000 with a 15% interest rate and 30 year term. Press **RUN**.

V = 40000	I = .15	N = 30	Answer: $505.777
at REDO (Y/N)?		input Y	
V = 50000	I = no entry		Answer: $632.222
REDO Y			
V = 60000	I = no entry		Answer: $758.666
REDO N			

If the buyer wishes to keep the monthly payment below $500, the program can be changed to read:

```
50IFP<500THEN20_
```

Test 3 mortgages to see if they meet the above criteria:

a. $40,000, 12%, 25 years
b. $30,000, 15%, 30 years
c. $50,000, 15%, 30 years

Mortgage "c" resulted in a payment in excess of $500 per month and therefore the program ended.

GOTO cycles a program to a previous or forthcoming statement line. In the mortgage example to continuously run the program, the lines 50 and 60 are replaced by:

```
50 GOTO 20
```

The **GOTO** statement can be used instead of **IF...THEN.**

BEEP produces an audible signal the number of times indicated after the command. Insert **BEEP** 2: in the last line of the mortgage program to indicate completion.

```
70 BEEP 2 : END
```

To limit the number of times a calculation can be performed or to number an operation a counter is used (looping). The most common form of counter is the LET statement:

$$I = I + 1$$

Each time this statement is passed in the program, the computer adds 1 to the previous value of I. Limit the number of times the mortgage calculation can be performed in any one run to 5. Add the following lines:

	Loop Characteristic:
12 A = 0	initialization
15 A = A + 1	incrementation

Change Line 50 to:

50 IF A < 5 GOTO 15	stop mechanism

FOR . . . TO, STEP, NEXT

FOR...TO in combination with **NEXT** repeats an operation the specified number of times. **STEP** is added if the incremental value is other than one. A **FOR-NEXT** loop must begin and end with the same variable.

Anatomy of a FOR statement

Command	Followed by	Explanation
FOR	variable	index
=	expression	initial value of index
TO	expression	final value of index
STEP	expression	increment

For 30-year mortgages ranging from \$10,000 to \$100,000 at an interest rate of 15%, calculate the monthly payment. Increment by \$10,000.

The maximum final value that the EL-5500Ⅱ can handle in a FOR ... to expression is $9.999999999 \times 10^{99}$. To alleviate the problem insert 1000 into Line 30 in the program. Rewrite the program until it reads:

```
10 PAUSE "MORTGAGE PAEMENT"
15 I=.15: N=30
20 FOR V=10 TO 100 STEP 10
30 P = 1000 * V/((1 - ((1 + I/12) ^ (-12 * N)))/(I/12))
40 WAIT 320: PRINT "PAYMENT $"; P
60 NEXT V
70 BEEP 2: END
```

Run the program. The answers are:
\$126.444, 252.888, 379.333, 505.777, 632.222, 758.666, 885.110, 1011.55, 1137.99, 1264.44

Nested looping is used if two or more variables are to be incremented. A **FOR-NEXT** loop must be completely inserted within another loop. Calculate the monthly payment on a 30-year mortgage in the amounts of \$30,000, \$60,000, and \$90,000 at interest rates of 14%, 15%, and 16%. The interest rate has been entered as a percent. To change to decimal, I is multiplied by .01 on Line 30. To rewrite, the ability to edit using the cursor is a special timesaver. On lines 15 and 20, just move the cursor through the line changing what needs to be changed, then press **ENTER**. The final program should read:

```
10 PAUSE "MORTGAGE PAYMENT"
15 N = 30
20 FOR V=30 TO 90 STEP 30
25 FOR I = 14 TO 16
30 P = 1000 * V/((1 – ((1 + .01 * I/12) ^ (–12 * N)))/(.01 * I/12))
40 WAIT 320: PRINT "PAYMENT $"; P
50 NEXT I
60 NEXT V
70 BEEP 2: END
```

Answer:

Mortgage Amount	Interest Rate	Montly Payment
$30,000	.14	$355.461
30,000	.15	379.333
30,000	.16	403.427
60,000	.14	710.923
60,000	.15	758.666
60,000	.16	806.854
90,000	.14	1066.38
90,000	.15	1137.99
90,000	.16	1210.28

USING

The answers to the mortgage problem have included more than two decimal places. **USING** formats an answer. It may be used by itself or with **PRINT**. A space must be left for the sign even if it is always positive. Since the answers for the mortgage problem were sometimes over $999, five spaces are required prior to the decimal point in the **USING** statement. After the decimal point, if only one digit would normally appear, a zero will be added. The third digit after the decimal

point is truncated. Add to the mortgage program:

```
35 USING "#####.##"
```

Change Line 40 to provide room for the answer:

```
40 WAIT 320: PRINT "PAYM. $"; P
```

If a particular USING format will be incorporated many times into a program, it is more efficient to use it as a string. At the beginning of the program include U$ = "####.##". In the program the statement becomes **USING** U$.

To avoid errors in a program eliminate a using format as soon as it is no longer needed by including the statement **USING** by itself.

The **USING** command can be placed within a **PRINT** statement. The complete format is:

PRINT (STRING, if any), **USING** (Format Expression); Variable

The comma is only required with a string.

GOSUB

Subroutines are a powerful programming tool. A portion of a program which may be repeated many times need be entered only once as a subroutine. Long programs are easier to read and complicated portions of a program can be moved around without having to be entered each time.

To return from a subroutine the command is **RETURN**. In the mortgage program set up the basic formula as a subroutine. As usual, key in a line number followed by **ENTER** to delete that line. Delete lines 15, 25, 50, 60.

Now key in: 20 **INPUT** V, I, N

Now use the arrows to get line 30, then move the cursor to the far left. Now press INS and change the line number to 500, delete "1000 * " from the program, and press **ENTER**. Now key in lines 30 and 600 until the program reads:

```
10 PAUSE "MORTGAGE PAYMENT"
20 INPUT V, I, N
```

```
30 GOSUB 500
40 WAIT 320: PRINT "PAYM. $"; P
70 BEEP 2: END
500 P=V/((1 - ((1 + .01 * I/12)^(-12 * N)))/(.01 * I/12))
600 RETURN
```

CLEAR, REM

CLEAR is used at the beginning of a program to insure that all variables are reset to zero and adequate space is cleared in memory for arrays. **CLEAR** also helps provide storage areas during a program when space is limited.

REM stands for remarks. Use it to make notes in a program. Items under **REM** will not appear when the program is run, but will appear in the program list.

Add these two commands to the mortgage program:

```
10 CLEAR : PAUSE "MORTGAGE PAYMENT"
15 REM PUT IN VARIABLES
25 REM CALCULATE PAYMENT
        rest of program remains the same
```

DATA, READ, RESTORE

Two commands are necessary for placing data into a program: **DATA** and **READ**. The **DATA** statement can be placed at any point in a program before **END**. Each item in the statement is separated by a comma. The grouping of the data is controlled by the **READ** statement. **DATA** statements can be broken into several lines.

The **READ** statement reads the data in the order of the variables as listed within the statement.

Calculate the monthly payment for 3 mortgages:

The program is:

```
10 CLEAR: PAUSE "MORTGAGE PAYMENT"
20 READ V, I, N
```

```
30 GOSUB 500
35 USING "#####.##"
40 WAIT 320: PRINT "PAYM. $"; P
50 GOTO 10
60 DATA 30000, 16, 30, 40000, 15.5, 25, 50000, 14.9, 22
70 BEEP 2: END
```

The subroutine program remains the same.

Run the program. The answers are: $403.42, $527.89, $645.67

The EL-5500II will indicate an "out-of-data" error when all the data has been used. This error can be avoided by placing a "flag" in the program tied to an **IF...THEN** statement. The flag acts as a test of the data and should be a value significantly different in form from the other data being read. For example, if all values are positive, –1 is a good flag. In the example above, add to the data line: –1, 0, 0 (null data must be included for all the variables):

```
25 IF V=-1 THEN GOTO 70
```

A choice of different sets of data is the function of **RESTORE**. Three sets of mortgage data are used by a broker. The adjusted program allows the user to make a choice by indicating the number.

PRINCIPAL:	INTEREST RATE:	TERM:
30,000	14, 15, 16	30, 25, 20
40,000	15, 16, 16.5	30, 28, 25
50,000	15.5, 16, 17	35, 30, 28

Program:

```
10 CLEAR
20 INPUT "SET 1, 2, 3?"; A
30 IF A=1 RESTORE 110: GOTO 60
40 IF A=2 RESTORE 120: GOTO 60
50 IF A=3 RESTORE 130: GOTO 60
```

```
 60 BEEP A: READ V, I, N
 65 IF V=-1 GOTO 140
 70 GOSUB 500
 80 USING "#####.##"
 90 WAIT 320: PRINT "PAYM. $";P
100 GOTO 60
110 DATA 30000, 14,30, 30000, 15,25, 30000, 16,20,-1,0,0
120 DATA 40000, 15,30, 40000, 16,28, 40000, 16.5,25,-1,0,0
130 DATA 50000, 15.5,35, 50000, 16,30, 50000, 17,28,-1,0,0
140 END
```

The subroutine program remains the same.

ON

The **ON** statement provides a means of combining several **IF...THEN** statements into one. The expression is followed by **GOTO** or **GOSUB** and a series of line numbers. In an **ON** statement the computer will **GOTO** the first of the line numbers if the expression is one, the second if it is two, and so forth.

An example of **ON...GOTO** appears in the HEAD example in Chapter 4.

DEF, AREAD

The letters on the lower two lines of the EL-5500III keyboard will bring forth specific programs when used to indicate a particular program. The use of **RUN** is eliminated. Define the mortgage program as **M** by changing the first program line to:

```
10 "M": CLEAR: PAUSE "MORTGAGE PAYMENT"
```

To use the program in the RUN mode press **DEF** and **M**.

A defined program can have the initial data entry made with the **AREAD** command. This command must appear in the first line of the program.

Calculate the monthly payment for 30-year mortgages at 15% for principal values of 35, 000, 40,000, 70,000, and 80,000. Change the program to:

```
10 "M": AREAD V
20 I= 15: N=30
30 GOSUB 500
40 USING "#####.##"
50 WAIT 320: PRINT "PAYM. $";P
60 BEEP 2: END
```

The subroutine program remains the same.

Change to RUN mode.

KEY IN:		*ANSWER:*
35000	DEF M	$442.55
40000	DEF M	505.77
70000	DEF M	885.11
80000	DEF M	1011.55

DIM, CONCATENATION

An array is a set of values named by one letter with a changing subscript. The number of values in the array is called its dimension. The subscript is called the index and a two-dimensional array will have two indices.

The use of arrays on the pocket computer should be limited due to the relatively small memories of these devices. The command **DIM** is used to create an array. The maximum size of the index in an array is 255. Zero is counted in an index, e.g., **DIM** C$ (3) sets up four reserved spaces.

If no size is specified for an index, 16 characters are automatically reserved for each item. The maximum number of characters is 80. Specifiying the size, the command is:

```
DIM C$*20
```

The user of the mortgage program wants to create a data base indicating the distance from the center of the city of five communities. Each item of data in the array will contain seven characters:

```
10 "A": DIM B$(4)*7
```

```
20 FOR I=0 TO 4
30 INPUT B$ (I)
40 NEXT I
50 FOR I=0 TO 4
60 PRINT B$ (I)
70 NEXT I
80 GOTO 50
```

To place the data into the computer go to the **RUN** mode and press DEF A. Each time ? appears type the entry and press **ENTER**. In this example:

? TROY-14	**ENTER**	? HEMP-18	**ENTER**
? NCITY28	**ENTER**	? NROC-15	**ENTER**
? ORNG-32	**ENTER**		

The first data item appears automatically after the last entry. Each time **ENTER** is pressed, each item will be displayed on the screen.

Multidimensional **DIM** statements are created by adding commas and expressions within the parenthesis. An excellent example of this technique is contained in the "spreadsheet" program in the financial part of the Applications chapter.

Concatenation is the combining of two or more strings into a phrase. The symbol + is always required and " " provides spacing.

A real estate broker places standard descriptions into a **DIM** statement to be combined to describe a property, including:

Age:	Older	New	
Bedrooms:	3-Bed	4-Bed	5-Bed
Type:	Ranch	Colonial	

Program:

```
10 DIM B$ (6)
20 FOR I=0 TO 6
30 INPUT B$ (I)
```

```
40 NEXT I
45 REM Dimension A$ (0) large enough to cover all combos.
50 DIM A$ (0) * 25
60 A$ (0) = B$(1)+" "+B$(2)+" "+B$(5)
70 PRINT A$(0)
80 END
```

Run the program keying in the 7 descriptive items. The values in Line 50 can be varied to meet each requirement. This A$ will read: NEW 3-BED RANCH.

AND, OR, NOT

AND, OR, and **NOT** are relational operators used with **IF...THEN** statements. Each is a logic test.

In the discussion of **IF...THEN** the following line was used:

```
50 IF P<500 THEN 20
```

Add the condition that the interest rate has to be less than 16%.:

```
50 IF P<500 AND I<.16 THEN 20
```

b. Either the payment must be lower than $500 or the interest rate below 15%:

```
50 IF P<500 OR I<.15 THEN 20
```

c. **NOT** reverses a condition

```
50 IF NOT P<500 THEN 20 (same as P>500)
```

LEN, LEFT$, MID$, RIGHT$

LEN indicates the number of characters contained in a string.

```
10 Z$ = "BOST, MA"
20 PRINT LEN (Z$)
```

Run the program. Answer is 7.

If a series of strings were to be analyzed by the first letter, add to the above program:

```
30 A$ = LEFT$ (Z$, 1)
40 PRINT A$
```

After the answer of 7, press **ENTER**. The answer will be B.

To add a listing of the states of each city in a hypothetical directory, use **RIGHT$**. The number in the parenthesis indicates the number of digits from the right the letters should be read.

```
50 S$ = RIGHT$ (Z$, 2)
60 PRINT S$
```

After the previous answer, press **ENTER** again. The answer is MA.

To alphabetize a series of strings a command that identifies internal letters in a string is valuable (**MID$**). The **MID$** expression has three parts: string identifier; starting point; number of characters.

```
70 X$ = MID$ (Z$, 2, 2)
80 PRINT X$
```

Press **ENTER** again. The answer is OS.

If a program user should answer "yes" or "no" to a question requiring a "y" or "n" reply, an error will occur. To reduce the chance of this type of error, the following program segment can be used:

```
50 INPUT "RERUN (Y/N)?"; Q$
60 IF LEFT$ (Q$, 1) = "Y" THEN (line #)
70 IF LEFT$ (Q$, 1) = "N" THEN (line #)
80 PAUSE "PLEASE ANSWER" : GOTO 50
```

Line 80 brings the user back to Line 50 if the answer to Line 50 does not begin with *Y* or *N*.

ASC, CHR$

Every character under the ASCII code has a decimal value. To count the number of times a letter occurs it is simpler to use the numeric value. To determine this value of the first letter of the string in the previous program:

```
90 PRINT ASC (Z$)
```

Pressing **ENTER** gives the value of 66.

In the RUN mode, key in:

```
CHR$ (66) ENTER
```

Answer: B

The **CHR$** statement converts an ASCII code number into its character equivalent.

STR$, VAL

STR$ turns a value into a string and **VAL** takes a value from a string. A number entered as a string will appear on the left side of the screen.

For a mortgage analysis it is necessary to produce the date of the last payment of the mortgage. The date is a string consisting of a three letter abbreviation for the month and a four-digit year. The following program adds the number of years of the term to the date.

```
10 INPUT B$
20 K = VAL RIGHT$ (B$, 4)
30 INPUT "TERM="; T
40 L = K + T
50 Q$ = LEFT$ (B$, 3)
60 C$ = Q$ + STR$ L
70 PRINT C$: END
```

If the mortgage begins in June of 1983 and runs for 30 years, the first input is JUN1983; TERM= is 30; and the final answer which will appear on the display is: JUN2013.

INKEY$

The **INKEY$** command is used to either speed up input or to limit the users choices. With **INKEY$** the input is limited to the key pressed and does not require the pressing of **ENTER**. The command takes the form:

```
10  A$ = INKEY$
```

RANDOM, RND

Random number generators primarily are used in statistical research and games. The **RANDOM** statement changes the beginning point for the generator each time to avoid duplication. The numbers are generated by the statement **RND** (*n*) with *n* being any number between 0 and 9.999999999 E 99.

A real estate office sold 200 houses during the past year. Select 10 sales at random to check the average sales level.

Program:

```
10  RANDOM
20  USING "####"
30  WAIT 150: PRINT RND (200)
40  GOTO 30
```

RUN. The first 10 numbers are:

84, 117, 89, 39, 78, 190, 151, 51, 153, 113

Try it. Do you get the same numbers?

PASS

The password system protects programs in the computer from modification and from unapproved copying. To protect the program or programs in memory the **PASS** command is followed by a character string which cannot exceed seven characters. To protect the previous program call it **MORTPAY** and key in:

PASS "MORTPAY" **ENTER**

The **LIST** command will not produce the statements in the program. To remove the password, key in the identical statement.

Debugging, TRON, TROFF

Errors in a program are called "bugs". One way of checking a program for problems is to use the trace mode. The mode is activated by keying in **TRON** and **ENTER** in **RUN** mode. The line number of each statement is displayed after execution of the statement. The next statement will not be performed until the ↓ key is pressed. To check the statement just executed press the ↑ key. The trace mode remains in effect until **TROFF** is keyed in.

Tape Recorder Commands

With the tape cassette interface, programs and data can be saved on standard or microcassette size tape recorders.

The command **CSAVE** places a program on cassette tape. By itself, **CSAVE** will transfer all programs in the EL-5500II memory to tape. Placing a comma after **CSAVE** followed by a name in quotations will make that name a password for the file being transferred to tape.

To identify one program from another on tape each can be assigned a file name. To transfer with a file name:

CSAVE "NAME"

Save the earlier mortgage payment program on tape under the name MORT (**CSAVE** "MORT"). When a file name is used a password can also be added as previously described.

CLOAD? will check the program in memory against the program as stored on tape. An error signal signifies that the program should be loaded on tape again.

The **CLOAD** command clears the computer's memory and loads the first program from the tape. **CLOAD** followed by a file name in quotes will result in the tape being searched for that file name and loading only that specific program.

CLOAD "MORT" will search for the mortgage program and load only it.

Data transfer to and from tape is accomplished with the statements **PRINT#** and **INPUT#**. To save the array in the **DIM** example on tape add the line:

PRINT# "DIST"; B$ (*)

Note: DIST is the file name. * indicates that all values of the array variable (B$) are to be saved.

Printer Commands

The printer adds a new dimension to pocket computing. Printing the mortgage payment program output allows a banker or real estate agent to hand an actual customized document to a potential customer. The basic command is **LPRINT.** All print commands following the line **PRINT=LPRINT** will activate the printer. To switch back insert the line **PRINT=PRINT.**

The mortgage payment program is expanded below to include a print-out of the initial information and an optional amortization schedule. Each month:

B = Interest portion of payment = Remaining principal * Monthly interest rate = R * I

G = Principal portion of payment = Monthly payment – Interest portion of payment = P – B

R = Remaining principal after payment = Previous principal – Principal portion of payment = R – G

Program:

```
10 "A": CLEAR : PAUSE "MORTGAGE PAYMENT": USING
15 REM INITIAL VALUES
20 INPUT "V=";V, "I="; I, "N="; N
25 REM CHANGE I TO MONTHLY BASIS, CALCULATE PAYMENT
30 I=I/12: P=V/((1-((1+I)^(-12*N)))/I)
40 PRINT = LPRINT
50 PRINT "PRINCIPAL $"; V
60 PRINT "INTEREST RATE "; 1200*I; "%"
70 PRINT "NO. OF YEARS="; N
80 USING "######.##"
90 PRINT "MONTH PAYM. $"; P
100 USING
110 INPUT "FULL SCH. (Y/N)?"; C$
120 IF C$="Y" THEN 200
130 END
200 REM INITIAL PRINCIPAL VALUE IS V
210 R=V
220 J=J+1
230 B=R*I: G=P-B: R=R-G
240 PRINT USING; "PERIOD#" ;J
250 PRINT "INTEREST PAID $";USING "######.##";B
260 PRINT "PRIN. PAID $"; G
270 PRINT "REMAINING PRI. $";R
```

```
280 REM TEST NUMBER OF PERIODS
290 IF J<12*N THEN 220
300 END
```

Run the program by pressing **DEF A** in the RUN mode.

		Key in:
V	=	70000
I	=	.13
N	=	30
FULL SCH. (Y/N)		Y

```
PRINCIPAL $70000.
INTEREST RATE 13.%
NO. OF YEARS=30.
MONTH PAYM.$   774.33
PERIOD # 1.
INTEREST PAID $   758.33
PRIN. PAID $    16.00
REMAINING PRI.$ 69983.99
PERIOD # 2.
INTEREST PAID $   758.15
PRIN. PAID $    16.17
REMAINING PRI.$ 69967.81
PERIOD # 3.
INTEREST PAID $   757.98
PRIN. PAID $    16.35
REMAINING PRI.$ 69951.45
PERIOD # 4.
INTEREST PAID $   757.80
PRIN. PAID $    16.53
REMAINING PRI.$ 69934.92
PERIOD # 5.
INTEREST PAID $   757.62
PRIN. PAID $    16.71
REMAINING PRI.$ 69918.21
PERIOD # 6.
INTEREST PAID $   757.44
PRIN. PAID $    16.89
REMAINING PRI.$ 69901.32
PERIOD # 7.
INTEREST PAID $   757.26
PRIN. PAID $    17.07
REMAINING PRI.$ 69884.24
```

CHAPTER 4

APPLICATIONS

MATHEMATICS

A. Feet/Inches Conversions

Adding or subtracting dimensions given as feet and inches can be difficult and confusing. In many applications it is necessary to convert dimensions given in feet and inches to decimal.

The program converts all dimensions given in feet to their equivalent in inches. Entries in inches only require pressing 0 then **ENTER** when the prompt is for "Feet". Inches including fractions can be entered by using the following format: 5+5/8. When the prompt is for 'Feet", the total can be obtained by inputting =.

The answer appears on the display alternatively as decimal feet and as feet and decimal inches several times. The user can change the number of times the answer appear and the length of time on the screen to meet their particular preference.

Program:

```
10 "F": CLEAR: U$= "###.####"
20 A=0: B=0: R$= "F": S$="IN": T$="=": L=12
30 INPUT "ENTER FT."; F$: IF F$=T$ THEN 70
40 A=VAL (F$): BEEP 1: INPUT "ENTER IN."; B: BEEP 2
50 C=A * L+B+C: E=(A * L+B)/L
60 WAIT 100: PRINT A; R$; B; S$; T$
65 PRINT USING U$; E; R$: BEEP 3: USING: GOTO 30
70 IF C<0 LET C=-(ABS C): E=-C/L: A=-(INT(C/L)): B=-(C-
   (A * L): GOTO 90
80 E=C/L: A=INT (C/L): B=C-A * L
90 FOR K=0 TO 3: PRINT USING ;"TOTAL "; A; R$; B; S$; T$
100 PRINT "TOTAL"; T$; USING U$; E; R$: NEXT K
```

```
120 INPUT "REVIEW (Y/N)?"; Q$
140 IF LEFT$ (Q$, 1)="Y" THEN 90
150 IF LEFT$ (Q$, 1)="N" THEN 10
160 PAUSE "PLEASE ANSWER": GOTO 120
```

Add 8′3 1/4″ to 11″ to 10′8 1/4″.

Change to RUN Mode.

Key in: **DEF F**

8 **ENTER**

3+1/4 **ENTER**

0 **ENTER**

11 **ENTER**

10 **ENTER**

8+1/4 **ENTER**

= **ENTER**

Answer: 19.875 Feet or 19 Feet 10 1/2 inches

B. Right-Angle Triangles

A triangle that has a right angle can be solved with two knowns as follows:

1. One side and one angle
2. Two sides

The program asks for each side and then for each angle. Once the computer has received enough information, it will solve for the other unknowns in the given triangle without requesting additional data. The answers will appear in the display along with their assignments. Line 400 allows the user to review the answer once again. The program extensively uses the conditional **AND**.

Program:

```
10 "A": CLEAR: F=90: INPUT "ENTER SIDE A"; A
20 INPUT "ENTER SIDE B"; B
30 IF (A+B)>A AND (A+B)>B GOTO 100
40 INPUT "ENTER SIDE C"; C
```

```
45 IF (A+C)>A AND (A+C)>C GOTO 100
50 IF (B+C)>B AND (B+C)>C GOTO 100
60 INPUT "ENTER ANGLE X"; D
65 IF (A+D)>A AND (A+D)>D GOTO 100
70 IF (B+D)>B AND (B+D)>D GOTO 100
75 IF (C+D)>C AND (C+D)>D GOTO 100
80 INPUT "ENTER ANGLE Y"; E
100 IF C+D+E=0 THEN LET C=√(SQU A +SQU B): D=ATN (A/B):
    E=F−D: GOTO 300
110 IF B+D+E=0 THEN LET B=√(SQU C −SQU A): D=ATN (A/B):
    E=F−D: GOTO 300
120 IF A+D+E=0 THEN LET A=√(SQU C −SQU B): D=ATN (A/B):
    E=F−D: GOTO 300
130 IF A+B+D=0 THEN LET A=C * COS E: B=C * SIN E: D=F−E:
    GOTO 300
140 IF B+C+D=0 THEN LET B=A * TAN E: C=A/COS E: D=F−E:
    GOTO 300
150 IF A+C+D=0 THEN LET A=B/TAN E: C=B/SIN E: D=F−E:
    GOTO 300
160 IF A+B+E=0 THEN LET A=C * SIN D: B=C * COS D: E=F−D:
    GOTO 300
170 IF B+C+E=0 THEN LET B=A/TAN D: C=B/COS D: E=F−D:
    GOTO 300
180 IF A+C+E=0 THEN LET A=B * TAN D: C=B/COS D: E=F−D
300 PRINT "SIDE A="; A
310 PRINT "SIDE B="; B
320 PRINT "SIDE C="; C
330 PRINT "ANGLE X="; D
340 PRINT "ANGLE Y="; E
400 INPUT "REVIEW ANS. (Y/N)", Q$
410 IF LEFT$ (Q$, 1)="Y" THEN 300
420 IF LEFT$ (Q$, 1)="N" THEN 10
```

```
430 PAUSE "PLEASE ANSWER": GOTO 400
```

Solve for the following triangles:

a. Side A = 3 Side B = 4

b. Side C = 5 Angle Y = 60°

c. Side A = Side B = 4

Change to RUN Mode:

Key in:	DEF A	*Answer:*
A = 3	ENTER	
B = 4	ENTER	
	ENTER	Side A = 3 Side B = 4
	⋮	Side C = 5 Angle X = 36.8699
Review	N ENTER	Angle Y = 53.1301
A =	ENTER	
B =	ENTER	
C = 5	ENTER	
X =	ENTER	
Y = 60	ENTER	
	ENTER	Side A = 2.5 Side B = 4.33
	⋮	Side C = 5 Angle X = 30
Review	N ENTER	Angle Y = 60
A = 4	ENTER	
B = 4	ENTER	
	ENTER	Side A = 4 Side B = 4
	⋮	Side C = 5.6569 Angle X = 45
		Angle Y = 45

C. Quadratic Equations

The quadratic formula is used to solve quadratic equations when the roots cannot be factored. The equation to be solved is put in the form:

$$ax^2 + bx + c = 0$$

The solution of the roots takes the form:

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

The portion of the root equation under the square root sign, $b^2 - 4ac$, is called the discriminant. When the discriminant is positive, there are two roots. A zero value for the discriminant results in a single real root and a negative discriminant produces roots which are complex numbers.

A complex number is shown as $a + bi$. To solve for complex roots, the negative sign is removed from the discriminant and i is added to the second part of the answer. The value of i is $\sqrt{-1}$.

Program:

```
10 "A": CLEAR: REM QUADRATIC EQUATION
20 INPUT "A="; A, "B="; B, "C="; C
30 D=SQU B -4 * A * C: IF D<0 THEN 120
40 USING
50 PAUSE "ROOT 1="
60 PRINT (√D - B)/(2 * A)
70 PAUSE "ROOT 2="
80 PRINT (-B-√D)/(2 * A)
90 INPUT "AGAIN (Y/N)?";Q$
100 IF Q$="Y" THEN 10
110 END
120 REM ** FIRST PART/SECOND PART**COMPLEX NUMBER
130 N=-B/(2 * A): I=√ABS D/(2 * A)
135 USING "###.###"
140 PAUSE "ROOTS="
150 PRINT N; ";"; I; "I": GOTO 90
```

Calculate the roots of the following equations:

a. $2x^2 + 8x - 10$

b. $x^2 - .2x + .004$

c. $20x^2 + 9x + 3897$

d. $x^2 + x + 1$

Key in:	DEF Q	*Answer:*
a. 2	ENTER	
8	ENTER	
−10	ENTER	ROOT 1 = 1.000
		ROOT 2 = −5.000
b. 1	ENTER	
−.2	ENTER	
.004	ENTER	ROOT 1 = 0.17746
	ENTER	ROOT 2 = 0.02254
c. 20	ENTER	
9	ENTER	
3897	ENTER	ROOTS = −0.225 ± 13.957i
d. 1	ENTER	
1	ENTER	
1	ENTER	ROOTS = −0.5 ± 0.866i

D. Polynomials

The formula defining the basic polynomial p of degree n or less is:

$$p(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + \cdots + a_{n-1}x^{n-1} + a_nx^n$$

In the program the degree of the polynomial is N and the range of subscripts for the powers is from 0 to N. Let B(N) represent the value of each constant. The program is divided into two parts. The first **FOR...NEXT** loop places the constants into memory. The second part evaluates the polynomial, allowing the user to revaluate for different values of x.

Program:

```
10 "S": CLEAR: PAUSE "POLYNOMIALS"
20 INPUT "N="; N: DIM B(N)
30 FOR I=0 TO N
40 INPUT B(I)
50 NEXT I
60 INPUT "X="; X: P=0
70 FOR I=0 TO N
80 P=B(I)*X^I+P
90 NEXT I
100 PRINT "P="; P
110 INPUT "TRY NEW X(Y/N)?"; Q$
120 IF Q$="Y" THEN 60
130 END
```

Calculate p for the following equation for values of x from 1 to 10:

$$p(x) = 2x^4 - 8x^3 + 4x^2 - 2x + 1$$

Change to RUN Mode.

Key in:	DEF P		
N=	4	ENTER	
	1	ENTER	
	−2	ENTER	
	4	ENTER	
	−8	ENTER	
	2	ENTER	*Answer:*
X=	1	ENTER	−3

X	2	3	4	5	6	7	8	9	10
P	−19	−23	57	341	997	2241	4337	7597	12381

E. Series

The values of many functions are approximations made with series. A series has an infinite number of terms. It is impossible to calculate every term in a particular series. The approximation improves with the number of terms included in the calculation.

For different values of x find the value of the definite integral:

$$\int_0^x \sin y^2 dy = \frac{x^3}{3} - \frac{x^7}{7(3!)} + \frac{x^{11}}{11(5!)} - \frac{x^{15}}{15(7!)}$$

Program:

```
10 "S": CLEAR: PAUSE "SIN-Y-SQU": N=3: G=1: L=0
20 INPUT "X="; X
30 INPUT "NO. OF TERMS"; Y
35 REM **VALUE OF EACH TERM**
40 P= X^N / (N * FAC G)
50 N=N+4: G=G+2: L=L+1
60 REM **MAKE EVEN TERMS NEGATIVE**
70 IF L/2-INT (L/2)=0 THEN LET P=-P
80 S=S+P
90 IF L<=Y THEN 40
100 PAUSE "SIN-Y-SQU="
110 PRINT S
120 REM **TO CHANGE # OF TERMS**
130 INPUT "TERMS #(Y/N)?"; Q$
140 IF Q$="Y" THEN 30
150 END
```

Change to RUN Mode.

Key in: **DEF S** Number of terms and values of x

Answers:

# of terms	Value of X: 1	.567	.1	.01
1	3.095238095E–01	6.031284904E–02	3.333309523E–04	3.333333331E-07
2	3.102813853E–01	6.03143242 E–02	Same	Same
4	3.102683028E–01	6.031432154E–02		
8	3.102683017E–01	Same		

A definite integral commonly used in statistics is:

$$\int_0^x e^{-y^2}\,dy = x - \frac{x^3}{3} + \frac{x^5}{5(2!)} - \frac{x^7}{7(3!)}$$

Program:

```
10 "F": CLEAR: PAUSE "E–Y–SQU": N=1
20 INPUT "X="; X
30 INPUT "NO. OF TERMS"; Y
35 REM **VALUE OF EACH TERM**
40 P=X^N/(N * FAC L)
50 N=N+2: L=L+1
60 REM **MAKE EVEN TERMS NEGATIVE**
70 IF L/2–INT (L/2)=0 THEN LET P=–P
80 D=D+P
90 IF L<=Y THEN 40
100 PAUSE "E–Y–SQU="
110 PRINT D
120 REM **TO CHANGE # OF TERMS**
130 INPUT "TERMS # (Y/N)?"; Q$
140 IF Q$="Y" THEN 30
150 END
```

Change to RUN Mode.

Key in: DEF E	Terms:	*Answer*
X= 1	3	7.428571429E–01
	5	7.467291967E–01
	7	7.468228068E–01
	9	7.468241207E–01
	11	7.468241327E–01
X= 2	11	8.799081128E–01
	13	8.819118443E–01
	15	8.820712251E–01
	17	8.820809061E–01

F. Simpson's Rule

Simpson's Rule is a commonly used method of obtaining an approximation of the value of a definite integral. The area under a curve is divided into an even number of parts. The width of each part is given by:

$$\Delta X = \frac{b - a}{n}$$

and the area covered by the curve of the integral is:

$$\int_a^b f(x)\,dx = \frac{\Delta x}{3}(f_0 + 4f_1 + 2f_2 + 4f_3 + 2f_4 + \cdots + 4f_{n-1} + f_n)$$

f_0 is $f(x)$ with $x = a$
f_1 is $f(x)$ with $x = a + \Delta x$
f_2 is $f(x)$ with $x = a + 2\Delta x$
$\ddots$
f_n is $f(x)$ with $x = b$

The integral to be evaluated is placed on Line 60 within the parenthesis. To obtain the area under the curve only three numbers need to be entered: the start-

ing point on the x-axis (A), the ending point (B), and the number of parts (N) the area will be divided into.

Calculate the total value of the definite integral for:

$$f(x) = x^3 + 2x^2 - x + 2$$

Program:

```
10  CLEAR: PAUSE "SIMPSONS RULE"
20  INPUT "B="; B, "A="; A, "N="; N
30  D=(B-A)/N
40  FOR I=0 TO N
50  X = A + I * D
60  F = X ^ 3 + 2 * SQU X - X + 2
70  T = L/2 - INT (L/2)
80  IF T>0 AND L<N THEN LET P=4*F
90  IF T=0 AND L<N-1 THEN LET P=2*F
100 IF L=0 OR L=N THEN LET P=F
110 L=L+1: R=R+P
120 NEXT I
130 PAUSE "AREA="
140 PRINT D*R/3
150 END
```

Change to RUN Mode.

Key in: B = 5 A = 1 N = 8

Answer: AREA = 234.66667

G. Matrices

A matrix is a rectangular array of numbers. The horizontal lists are called rows and the verticals, columns. A particular element in a matrix is identified by row and column numbers:

A (row #, column #)

The following table is converted to a 3*3 matrix:

INVENTORY AIRPLANE PARTS

	BEASS	STEEL	COPPER
small	27	32	48
medium	82	51	77
large	15	28	19

MATRIX

$$A = \begin{vmatrix} 27 & 32 & 48 \\ 82 & 51 & 77 \\ 15 & 28 & 19 \end{vmatrix}$$

The first row and first column in a matrix are denoted as zero. Therefore the number 27 is A(0, 0) and 51 is A(1, 1). I is the usual designation for the row subscript and J for columns. The general term is:

A (I, J)

A matrix in BASIC is specified by a **DIM** statement.

Program:

```
10   DIM D(2,2)
20   FOR I=0 TO 2
30   FOR J=0 TO 2
40   READ D(I,J)
50   NEXT J: NEXT I
60   DATA 27, 32, 48
70   DATA 82, 51, 77
80   DATA 15, 28, 19
```

To generalize use R and C as subscripts. Because 0 is used on the EL-5500III, R and C will be one more than the actual number. To compensate, one is usually deducted in the **FOR** statement.

Program:

```
10   INPUT R,C
20   DIM D(R,C)
30   FOR I=0 TO R-1
40   FOR J=0 TO C-1
50   READ D(I,J)
60   NEXT J: NEXT I
70   DATA 27, 32, 48
80   DATA 82, 51, 77
90   DATA 15, 28, 19
```

Matrices may be added by adding corresponding elements assuming the matrices have the same number of elements and equal numbers of rows and columns.

An additional inventory of airplane parts is received with the following matrix:

$$B = \begin{vmatrix} 30 & 50 & 10 \\ 40 & 80 & 20 \\ 20 & 7 & 30 \end{vmatrix}$$

```
10   CLEAR: INPUT R,C
20   DIM D(R,C), B(R,C)
30   FOR I=0 TO R-1
40   FOR J=0 TO C-1
50   READ D(I,J)
60   NEXT J: NEXT I
70   FOR I=0 TO R-1
80   FOR J=0 TO C-1
90   READ B(I,J)
```

```
100 PRINT D(I,J) + B(I,J)
110 NEXT J: NEXT I
120 STOP
130 DATA 27, 32, 48, 82, 51, 77, 15, 28, 19
140 DATA 30, 50, 10, 40, 80, 20, 20, 7, 30
150 END
```

Change to RUN Mode

Key in: RUN

? 3 ENTER

? 3 ENTER

Answer:

$$C = \begin{vmatrix} 57 & 82 & 58 \\ 122 & 131 & 97 \\ 35 & 35 & 49 \end{vmatrix}$$

A matrix may be decreased by subtraction in the same manner as addition or the DIM B can be multiplied by −1. The latter case is called scalar multiplication where a matrix is multiplied by a real number called a scalar.

To multiply the second inventory by 2 in the above program add the following line:

```
105 PRINT 2 * B(I,J)
```

Matrix multiplication can only be carried out if the number of columns in the first matrix equals the number of rows in the second matrix. If D(R, C) is to be multiplied by B(W, L); C must be equal to W. The resulting product will have the dimensions R∗L; P(R, L).

Each value in the product is based on the equations:

$$P(0,0) = D(0,0) * B(0,0) + D(0,1) * B(1,0) + \cdots + D(0,C) * B(C,0)$$

$$\vdots \qquad\qquad\qquad\qquad \vdots$$

$$P(R,L) = D(R,0) * B(0,L) + D(R,1) * B(1,L) + \cdots + D(R,C) * B(C,L)$$

Program:

```
10  CLEAR: INPUT "R="; R, "C="; C, "W="; W, "L="; L
20  DIM D(R,C), B(W,L), P(R,L)
30  REM **CHECK C=W**
40  IF C<>W THEN 10
50  FOR I=0 TO R-1
60  FOR J=0 TO C-1
70  INPUT "DATA1="; D(I,J)
80  NEXT J: NEXT I
90  FOR I=0 TO W-1
100 FOR J=0 TO L-1
110 INPUT "DATA 2 = "; B(I,J)
120 NEXT J: NEXT I
130 REM **MULTIPLICATION**
140 FOR I=0 TO R-1
150 FOR J=0 TO L-1
160 REM *CLEAR P-SUMS TO ZERO FOR NEXT PROGRAM LOOP
170 P(I,J)=0
180 REM *THIRD INDEX K NEEDED FOR MATCHING SUBSCRIPT
    IN EACH TERM
190 FOR K=0 TO C-1
200 P(I,J) = P(I,J) + D(I,K) * B(K,J)
210 NEXT K
220 PRINT P(I,J)
230 NEXT J: NEXT I
240 END
```

Production of three fruit juice drinks requires the following ounces/point of real juice:

	TYPE OF DRINK			
	A	B	C	D
ORANGE	1	2	4	4
LIME	3	3	3	1
LEMON	2	1	2	1

Production anticipated for the next 3 weeks.

		WEEK		
		1	2	3
DRINK:	A	45000	60000	50000
(pints)	B	49000	22000	70000
	C	42000	46000	57000
	D	19000	20000	21000

Calculate the required ounces of each fruit juice per week.

Change to RUN Mode

Key in: RUN

R = 3 ENTER W = 4 ENTER
C = 4 ENTER L = 3 ENTER

DATA 1 = 1 2 4 4 3 3 3 1 2
1 2 1

DATA 2 = 45000 60000 50000 49000 22000
70000 42000 46000 57000 19000
20000 21000

Answers: Requirements (Ounces)

WEEK	ORANGE	LIME	LEMON
1	387000	168000	502000
2	378000	382000	482000
3	242000	254000	305000

In many applications of matrices the determinant of a square matrix must be calculated. The determinant of a 3∗3 matrix is given as:

$$\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix} = a_1 b_2 c_3 + a_2 b_3 c_1 + a_3 b_1 c_2 - a_1 b_3 c_2 - a_3 b_2 c_1 - a_2 b_1 c_3$$

Program:

```
10  CLEAR: DIM B(2,2)
20  FOR I=0 TO 2
30  FOR J=0 TO 2
40  INPUT "DATA ="; B(I,J)
50  NEXT J: NEXT I
60  G=B(0,0) * B(1,1) * B(2,2) + B(1,0) * B(2,1) * B(0,2) + B(2,0)
    * B(0,1) * B(1,2)
70  F=-B(0,0) * B(2,1) * B(1,2) - B(2,0) * B(1,1) * B(0,2) -
    B(1,0) * B(0,1) * B(2,2)
80  PRINT "D="; G+F
90  END
```

Calculate the Determinant for the matrix:

$$\begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix}$$

Change to RUN Mode

Key in: RUN

Data = 1 2 3 4 5 6 7 8 9

Answer: D = 0

Change the number 9 to 10

Answer: D = –3

STATISTICS

A. "T" Test

A small sample in statistics usually means less than 30. Small samples are particularly important where the data is very expensive or diffcult to obtain or a test requires destruction of a product.

The t-distribution is close to the normal distribution and permits checking of a hypothesis even with a small sample.

$$t = \frac{\bar{x} - \mu_0}{s/\sqrt{n}}$$

where $\bar{x}$ = sample mean

s = sample standard deviation

n = sample size

μ_0 = required population mean

The EL-5500III is particularly convenient for running "t" teste. Data input in the calculator mode is automatically stored in letter memories in the BASIC modes as follows:

LETTER	STATISTICAL DATA STORED
U	Σy^2
V	Σy
W	Σxy
X	Σx^2
Y	Σx
Z	n

The "t" test can be programmed using the store memories.

Let $A = \mu_0$

$$t = \frac{\bar{x} - \mu_0}{s/\sqrt{n}} = \frac{\frac{\Sigma x}{n} - \mu_0}{\sqrt{\frac{\Sigma x^2 - n(\Sigma x/n)^2}{n(n-1)}}} = \frac{Y/Z - A}{\sqrt{(X - Z(Y/Z)^2)/Z(Z-1)}}$$

Program:

```
10   PAUSE "T–TEST"
20   INPUT "POP.MEAN="; A
30   T = (Y/Z–A)/√((X–Z * SQU(Y/Z))/Z * (Z–1))
40   USING "##.###^"
50   PRINT "T="; T
```

A sample of 5 transistors are used until they fail. The life (hours) of each is 7400, 8000, 8300, 8700, 9900. A buyer of transistors requires a mean-life of 8100 hours. Will the buyer be safe with a 5% level of confidence if these transistors are purchased?

Change to Calculator Mode:

Key in: **STAT**

7400 **DATA** 8000 **DATA** 8300 **DATA** 8700 **DATA** 9900 **DATA**

Change to RUN Mode

Key in: **RUN**

POP. MEAN = 8100 **ENTER**

Answer: T = 2.153 E – 01

From the table, the value for t at a 5% confidence level is 2.13. We use the sample size (n) –1 as the degrees of freedom.

Since the t calculated at .2153 is less than 2.13, the buyer can conclude that the transistors purchased could be from a population with a mean lifetime equal to or greater than 8100 hours 95% of the time.

B. Chi-square Test

The x^2 Test is used to determine if a set of data when compared to an ideal or predetermined distribution of that data has a variance from probability or some preset expectation. The formula for the test is:

$$x^2 = \Sigma \frac{(O-E)^2}{E}$$

where O = Observed Frequency
E = Expected Frequency

The hypothesis is rejected if the x^2 statistic falls in the rejection region of the x^2 distribution.

The proportion of adult females (over age 21) in society is 52% of total adults. The number of female workers earning above $20,000/year in three companies was established. Analyzing the data, is the difference between the actual proportions compared to the expected frequency due to chance.

EXPECTED FREQUENCY OF EMPLOYEES BY SEX

RUN Mode

Key in: A = .52

COMPANY	TOTAL		FEMALE	MALE
A	10000 * A	ENTER	5200	4800
B	5650 * A	ENTER	2938	2712
C	3290 * A	ENTER	1711	1579

The table below indicates the actual number with the expected below in parenthesis. Each cell will be calculated in the x^2 calculation.

	COMPANY		
	A	B	C
Female	3100	2900	1800
	(5200)	(2938)	(1711)
Male	6900	2750	1490
	(4800)	(2712)	(1579)
Total	10000	5650	3290

Program:

```
10  CLEAR: L=0: PAUSE "CHI-SQUARE"
20  INPUT "CELLS ="; N
30  INPUT "OBS ="; O, "EXP ="; E
40  G = SQU (O-E)/E
50  H=G+H: L=L+1
60  IF L<N THEN 30
70  USING "#####.##"
80  PRINT "CHI-SQU="; H
90  END
```

Change to RUN Mode

Key in:

CELLS = 6 **ENTER**

OBS = 3100 2900 **ENTER**, etc.

EXP = 5200 2938 **ENTER**, etc.

Answer: CHI–SQU = 1777.49

The $\chi^2_{0.05}$ column in the table represents the dividing line which cuts off 5% of the right tail of the distribution. The degrees of freedom is one less than the number of sample proportions being tested.

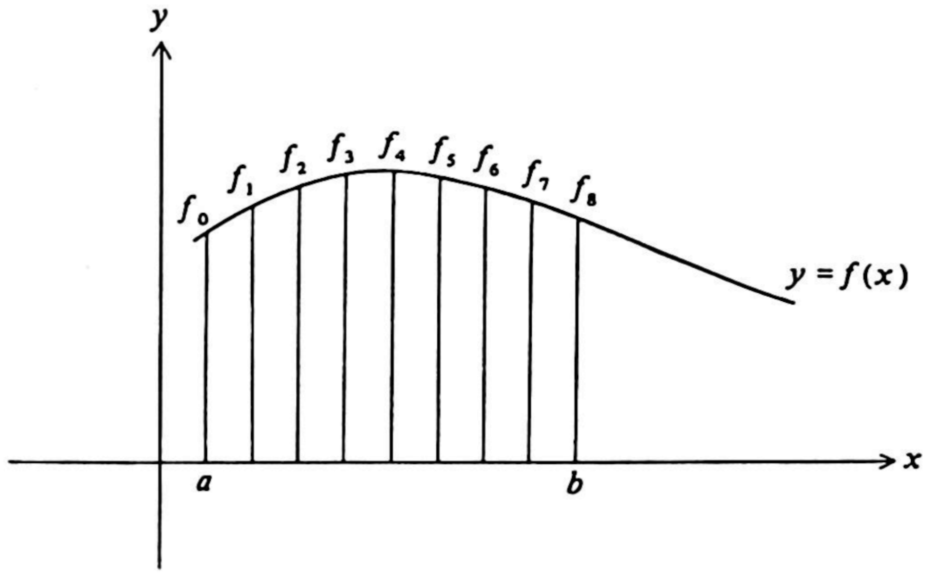


Figure 4.1 Curve-Simpsons rule

In the above example the *df* is 2 and $x^2_{0.05} = 5.991$. The value obtained is significantly higher indicating the difference between what was expected and what was actually happening was due to factors other than chance.

BUSINESS

A. Break-Even Analysis

"Break-even" is a business technique used to determine the quantity of a product where total costs will equal variable costs plus fixed costs. Fixed costs are depreciation, light, power, utilities, rent, insurance, administrative salaries and others. Variable costs are materials and labor. The number of units required to reach the break-even point is:

$$\text{Quantity} = \frac{\text{Fixed Cost}}{\text{Price} - \text{Variable cost/unit}}$$

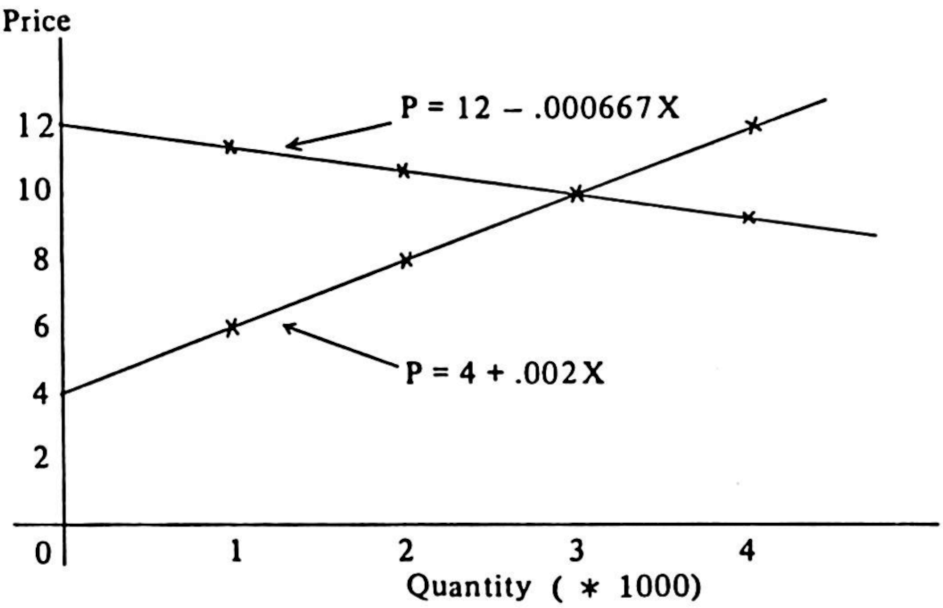


Figure 4.2 Supply-Demand Curves

It is possible to use supply and demand curves to determine the price. Supply curves measure the willingness of manufacturers to provide product at different prices. Demand curves reflect the willingness of consumers to buy at different prices.

The supply and demand curves can alternatively be used to determine the price in the break-even equation. The basic equation for these curves is P = A + BX.

F = Fixed Cost
P = Price
V = Variable cost/unit

Program:

```
10  "A": CLEAR: PAUSE "BREAK-EVEN ANAL"
20  PAUSE "PRICE CALC"
30  INPUT "A ="; A, "B="; B
40  INPUT "F.C.="; F, "V.C. ="; V
50  INPUT "X="; X
55  USING
60  P=A+B * X
70  PRINT "P="; P
```

```
80   REM BREAK–EVEN ANALYSIS
90   W = F / (P – V)
100  Z = W – X
105  USING "#####."
110  PRINT "AMT. ="; W
115  PRINT "DIFF. ="; Z
120  IF Z < –1 OR Z > 1 THEN GOTO 50
130  END
```

The supply curve for a particular product is: $p = 4 + .002X$. The demand curve is: $P = 12 - .000667X$. The fixed cost is \$10,000 and the variable cost is \$6.00. Determine break-even quantities based on the supply and demand curves. Use supply and demand quantities of 2,000 in each case.

Change to RUN Mode

Key in:	DEF A	NOTES
A =	4 ENTER	Using Supply Curve
B =	.002 ENTER	
F.C. =	10000 ENTER	
V.C. =	6 ENTER	
X =	2000	Quantity

Answer:

P =	8	Price
AMT. =	5000	Break-even Quantity
DIFF =	3000	Difference between break-even quantity and supply curve quantity.

Key in: DEF A

A =	12 ENTER	Using demand curve
B =	–.000667 ENTER	
F.C. =	10000 V.C. = 6 X = 2000	

Answer:

P = 10.666	Price
AMT. = 2143 DIFF = 143	Difference between break-even quantity and demand curve quantity
X = 3000	
P = 9.999	
AMT. = 2500 DIFF = –500	Demand quantity at $10 is 500 more than required to break-even

Retry with Supply Curve

Key in: **DEF A**

A = 4 B = .002 F.C. = 10000 V.C. = 6

X = 3000 **ENTER** (New Value)

Answer:

P = 10 AMT = 2500 DIFF = –500

(same answer as for Demand Curve)

This is the equilibrium point of the two curves. When the equilibrium price is higher than required to break even, the investment situation is favorable.

B. Loans

Amortization of a loan based on an equal payment each period is described in Chapter 3. Some loans require each payment to include an equal amount for principal. The payment will change each period.

The interest rate each month is based on a 365-day year: INTEREST RATE = (ANNUAL RATE/365) * 30. This calculation is carried out on line 80 of the program. The annual rate (line 50) must be entered as a decimal. This example must be used with CE-126P interface. Users without the interface should replace any LPRINT with PRINT.

Program:

```
10   CLEAR: LPRINT "MONTHLY PAYMENT"
20   LPRINT "EQUAL PRIN/MON"
30   INPUT "MONTH ="; E, "DAY ="; D, "YEAR ="; H
40   INPUT "PRINC.="; P
50   INPUT "INT. (.XX) ="; I
60   INPUT "YEARS="; Y
70   LPRINT "DATE: "; E; D; H
80   LPRINT "12 PAYMENTS/YEAR"
90   LPRINT "PRINC. $"; P
100  LPRINT "INT.RATE"; 100 * I; "%"
110  LPRINT "TERM="; Y; "YEARS"
120  N=12 * Y: R=(I/365) * 30: X=P/N: B=P
130  K=E
140  C=B * R
150  M=X+C: B=B-X
160  L=L+1: K=K+1
170  IF K=1 THEN LET H=H+1
180  PRINT = LPRINT
190  PRINT "#"; L; "   "; K; "   "; K; D; H
200  USING "######.##"
210  PRINT "INT. = $"; C
220  PRINT "PRIN.=$"; X
230  PRINT "PAYM.=$"; M
240  PRINT "BAL. =$"; B
250  IF K=12 THEN LET K=0
260  USING
270  IF L<N THEN GOTO 140
280  END
```

Calculate the monthly payments for a $50,000 loan for 3 years at 15%.

Change to RUN Mode.

Key in:

MONTH = 6 DAY = 25 YEAR = 1983 PRIN = 50000 INT = 1.5 YEARS = 3

Answers:

```
MONTHLY PAYMENT
EQUAL PRIN/MON
DATE:6.25.1983.
12 PAYMENTS/YEAR
PRINC.$50000.
INT.RATE15.%
TERM=3.YEARS
#1. 7. 7.25.1983.
PRIN.=$  1388.88
PAYM.=$  2005.32
BAL.=$ 48611.11
#2. 8. 8.25.1983.
PRIN.=$  1388.88
PAYM.=$  1988.20
BAL.=$ 47222.22
#3. 9. 9.25.1983.
PRIN.=$  1388.88
PAYM.=$  1971.08
BAL.=$ 45833.33
```

C. Days Between Dates

The actual number of days between two dates is important for many types of financial calculations.

The program uses a preset reference date. Leap years are included with the proper number of days.

Program:

```
10  "A": CLEAR
15  REM STARTING DATE
20  INPUT "FROM YEAR"; R, "MONTH ="; S, "DAY ="; T
30  INPUT "TO YEAR"; U, "MONTH="; V, "DAY="; W
40  REM **REFERENCE YEARS**
50  H=R
60  G=S: I=T
```

```
70   GOSUB 500
80   J=I
100  H=U
110  G=V: I=W
115  GOSUB 500
120  X=I-J
130  PRINT "DAYS"; X
140  GOTO 30
500  IF G-3>=0 LET Z=-(G-3) * 30.6-.5: GOSUB 600: I=I-Z:
     GOTO 530
510  H=H-1
520  Z=(-(G-3)-12) * 30.6-.5: GOSUB 600: I=I-Z
530  Z=H * 365.25: GOSUB 600: I=I+Z
540  Z=H/100: GOSUB 600: I=I-Z
550  Z=H/400: GOSUB 600: I=I+Z
560  I=I-307: RETURN
600  X=INT ABS Z: Z=SGN Z * X: RETURN
```

Change to the RUN Mode.

Calculate the number of days in 1983 and from 5/20/80 to 5/22/89.

Key in: **DEF A**

a. FROM YEAR 1983 **ENTER** TO YEAR 1983 **ENTER**
MONTH = 1 **ENTER** MONTH = 12 **ENTER**
DAY = 1 **ENTER** DAY = 31 **ENTER**

Answer: DAYS 364

b. FROM YEAR 1980 **ENTER** TO YEAR 1989 **ENTER**
MONTH = 5 **ENTER** MONTH = 5 **ENTER**
DAY = 22 **ENTER** DAY = 22 **ENTER**

Answer: DAYS 3287

D. DISCOUNTED CASH FLOW

Discounted Cash Flow (DCF) provides the average rate of compound interest earned by a project on money invested over the life of the project. DCF discounts the annual cash flows to properly reflect the time value of money on a continuous compound interest basis. The rate of return and net present value of a DCF are based on the cash flow from earnings for each period. The cash flow is usually the sum of the net profit and depreciation. Any sales of land and equipment at the end of the project are included in the cash flow for the last period.

If the investor has a minimum return to achieve on an investment, then the Net Present Value (NPV) technique is used. If the NPV is positive, the project goal has been achieved. If negative, profit objectives have not been met. The NPV or future cash flows is represented by the series:

$$\mathrm{NPV} = -\mathrm{INV} + \frac{C_1}{(1+i)^1} + \frac{C_2}{(1+i)^2} + \cdots + \frac{C_n}{(1+i)^n}$$

where: INV = initial investment

CN = cash flow for a period

i = rate of return

The Internal Rate of Return (IRR) uses the same formula as NPV with NPV equal to 0. The actual IRR is arrived at by iteration, taking successive guesses at the value of i until the NPV is near 0. The program provides an automatic cutoff in Line 180 if the amount is within 10% of the original NPV. The **DIM** statement on line 30 provides for an unlimited number of cash flow periods within the capacity of the computer. The initial investment is a negative cash flow and is entered as C(0).

Program:

```
10  "A": CLEAR
20  INPUT "NO. OF PERIODS"; N, "RATE"; R
30  DIM C(N)
40  FOR I=0 TO N
50  INPUT "CASH FLOW"; C(I)
```

```
60   REM ** NPV FOR EACH PERIOD**
70   P=C(I)/(1+R)^I
80   T=T+P
90   NEXT I
95   USING "#######.##"
100  PRINT "NPV="; T
110  F=0: G=0: INPUT "IRR="; R
120  FOR I=0 TO N
130  F=C(I)/(1+R)^I
140  G=F+G
150  NEXT I
160  PAUSE "EST NPV="
170  PRINT G
180  IF G>T*.1 OR G<-T*.1 GOTO 110
190  PRINT "IRR="; 100*R; "%"
```

An investor wishes to make a five year investment in a piece of property which can be used as a parking lot. The property will cost \$200,000 and is expected to appreciate in value at the rate of 10% per year. The investor wants an annual rate of return of 20%. The net cash flow for the parking lot is:

YEAR	1	2	3	4	5
AMOUNT	\$30,000	45,000	24,000	52,000	40,000

Will the investor achieve a 20% rate of return and what is the actual IRR?

To determine the value of the land in 5 years, use the future value formula:

$$FV = PV\ (1+i)^n = 200{,}000\ (1+.1)^5$$

RUN Mode

Key in:

2 EXP 5 * 1.1 ^ 5 + 40000 ENTER

Answer:	362102		
Key in:	DEF A		NOTES
	PERIODS = 5 ENTER		
	RATE = .2 ENTER		
	CASH FLOW = −200000	ENTER	Initial Investment
	CASH FLOW = 30000	ENTER	First Period
	45000	ENTER	Second
	24000	ENTER	Third
	52000	ENTER	Fourth
	362102	ENTER	Fifth Period Plus Resale Value of Land
Answer:	NPV = $40736.72 ENTER		
Key in:	IRR = .3		30% estimate for IRR
Answer:	EST NPY = −23640.62 ENTER		
	IRR = .25		TRY 25%
	EST NPV = 5040.78 ENTER		
	IRR = .26		TRY 26%
	EST NPV = −1197.76 ENTER		
Answer:	IRR = 26% The correct IRR is close to 26%.		

E. Spread Sheet

A spread sheet allows the inclusion of financial or other data in a table and the preparation of totals of the columns and numbers.

The EL-5500III has a limited capability for spread sheet problems because of a relatively small memory. The program is included for smaller problems and for its educational value.

The last section of the program called "Print Module" repeats all the inputs and the cross totals. The data is placed into the computer reading down the columns.

The program is in four sections. The first is for entering the data. The columns

and rows are totaled in the second and third parts. The last part of the program is a print module which lists each of the data inputs for checking and gives the totals for the columns and the rows. The number of elements are entered with the columns (E) first, followed by the rows (F).

Program:

```
5    CLEAR: PAUSE "SPREAD–SHEET"
10   INPUT "ELEMENTS"; E, F
20   DIM B(E, F)
30   REM **INPUT DATA**
40   FOR X=0 TO E–1
50   FOR Y=0 TO F–1
60   INPUT "ENTER DATA"; B(X, Y)
70   NEXT Y: NEXT X
100  FOR X=0 TO E–1
110  FOR Y=0 TO F–1
120  B(E,Y) = B(E,Y) + B (X,Y)
130  NEXT Y: NEXT X
140  FOR Y=0 TO F–1
150  P=B(E,Y) + P
160  FOR X=0 TO E–1
170  B(X,F) = B(X,F) + B(X,Y)
180  NEXT X: NEXT Y
200  REM **PRINT MODULE**
210  FOR X=0 TO E
220  FOR Y=0 TO F
225  IF X=E AND Y=F GOTO 245
230  PRINT B(X,Y)
240  NEXT Y: NEXT X
245  PAUSE "GRAND TOTAL ="
250  PRINT P: END
```

Calculate the cross-totals and the grand total for the following information:

BOND SALES

TYPE	January	February
Industrial	282	342
Municipal	181	720
Government	410	500

Change to RUN Mode

Key in:	RUN	ENTER
	ELEMENT	2 ENTER
	?	3 ENTER
	ENTER DATA	282 181 410 342
		720 500

Display (Press **ENTER** for each number):

			(totals)
	282	342	624
	181	720	901
	410	500	910
(totals)	873	1562	2435

PHYSICS

A. Orbitting

In satellite motion, the orbitting of one object around another, the effects of acceleration and the gravitational force must be taken into consideration. The gravitational force is an inverse square central force. This continuous force acting on a satellite in orbit changes its magnitude and direction as the sattelite moves.

The orbit can be approximated by breaking the satellite's flight into a series of small steps.

A comet has an initial distance from the sun of 4 Astronomical Units (A.U.) and an initial velocity of 2 A.U./year. Plot the orbit.

The applicable parameters are:

Orbital Period = 2π = 6.28 = time units for one circumference

G = universal gravitational constant

M = mass of the central body

C = –GM = –1 = proportionality constant which is negative when the central force is attractive

n = 2 = dependence on center-to-center distance by the central body and the body moving by it.

Δt = 60 days = 60/365 * 6.28 = 1.033

r = distance from the central body

a = acceleration

V = velocity

Vyo = 2 A.U./6.28 = 0.3183

x = 4 A.U. = initial distance from the sun

The applicable equations are:

$$\text{Distance} = r = \sqrt{x^2 + y^2}$$

$$\text{Acceleration} = a_x \Delta t + Cr^{-(n+1)} x \, \Delta t$$
$$a_y \Delta t + Cr^{-(n+1)} y \, \Delta t$$

$$\text{Velocity} = \text{New } V_x = \text{Old } V_x + a_x \Delta t$$
$$\text{New } V_y = \text{Old } V_y + a_y \Delta t$$

* Based on material from the book: COUNTDOWN ON PROGRAMMABLE CALCULATORS, by R. Eisberg & W.Hyde, Dilithium Press, Portland, OR, 1979.

$$\text{Location} = \text{New } x = \text{Old } x + V_x \Delta t$$
$$\text{New } y = \text{Old } y + V_y \Delta t$$

Program:

```
10  CLEAR
20  C=-1: N=2: T=1.033
30  INPUT "XO ="; X, "YO="; Y
40  INPUT "VO ="; V, "VYO ="; W
50  R = √ (SQU X + SQU Y)
60  V = V + C * R ^ -(N+1) * X * T
70  W = W + C * R ^ -(N+1) * Y * T
80  X = X + V * T : Y = Y + W * T
90  USING "##.##"
100 PRINT "X ="; X; "Y ="; Y
110 GOTO 50
```

Change to RUN Mode

Key in: **RUN**

XO = 4   YO = 0   VO = 0   VYO = .3183

Answers: (Use to plot on an X–Y Axis)

Press **ENTER** each time for a new set of X and Y coordinates

X =	Y =
3.93	0.32
3.79	0.65
3.59	0.96
3.31	1.25
2.95	1.51
2.50	1.73
1.96	1.88
1.32	1.93
0.56	1.82
–0.27	1.43
–1.02	0.55
–1.08	–0.69

This orbit will fail to close when plotted on graph paper. The reader should try a smaller Δt (20 days) to get a perfect ellipse.

B. Moment of Inertia

The moments of inertia about any x and y axes for n number of elements are I_{xx}, and I_{yy}, and I_{xy}. The program transfers each of these inertias to the $x_{c.g.}$ and $y_{c.g.}$ axes of the element set.

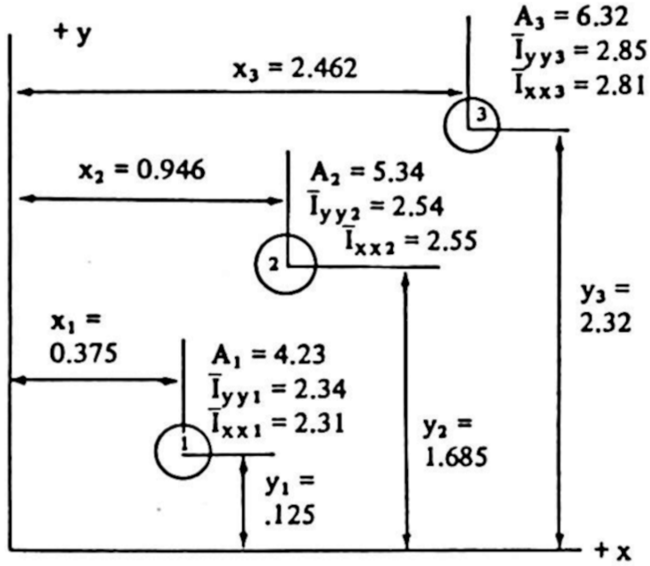


Figure 4.3 Moment of Inertia

The input information required is:

1. x and y distances to each element
2. corresponding element sizes
3. element $\Sigma \bar{I}_{xx}$ and $\Sigma \bar{I}_{yy}$

The applicable formulas are:

1. $\bar{x} = \dfrac{\Sigma A x}{\Sigma A}$ 2. $\bar{y} = \dfrac{\Sigma A y}{\Sigma A}$

3. $I_{xy} = \Sigma A x y - \bar{y}\,\Sigma A x - \bar{x}\,\Sigma A y + \bar{x}\,\bar{y}\,\Sigma A$

4. $I_{yy} = \Sigma A x^2 + \Sigma \bar{I}_{yy} - (\Sigma A y)^2/A$

5. $I_{xx} = \Sigma A y^2 + \Sigma \bar{I} - (\Sigma A x)^2/A$

6. Radius of Gyration:

$$RHO_{xx} = \sqrt{I_{xx}/A} \qquad RHO_{yy} = \sqrt{I_{yy}/A}$$

Where:

I_{xx} = moment of inertia about the x-axis

I_{yy} = moment of inertia about the y-axis

I_{xy} = product of inertia about x and y axis

n = number of elements

$c.g.$ = center of gravity or centroid of the element set

$\Sigma\bar{I}_{xx}$ = the total for all the elements of each element moment of inertia about its own $x_{c.g.}$ axis

$\Sigma\bar{I}_{yy}$ = similar to $\Sigma\bar{I}_{xx}$ but about $y_{c.g.}$ axis

The program uses **DIM** statements to store and manipulate the data. The **DIM** statement in Line 20 has one variable, E, which represents the number of elements. The number of data points for each element will always be the same (5).

Within the program:

Let $L = \Sigma Ay$ $\qquad P = \Sigma Ax^2$

$M = \Sigma Ax$ $\qquad Q = \Sigma Ay^2$

$N = \Sigma Axy$

Program:

```
10  CLEAR : PAUSE "MOMENTS"
20  INPUT "ENTER ELEMENTS"; E
30  DIM B(E,4)
40  REM **INPUT DATA**
50  FOR X=0 TO E-1
60  FOR Y=0 TO 4
70  INPUT "DATA="; B(X,Y)
80  NEXT Y: NEXT X
90  REM *TOTAL COLUMNS**
100 FOR X=0 TO E-1
```

```
110 FOR Y=0 TO 4
120 B(E,Y) = B(E,Y) + B(X,Y)
130 NEXT Y: NEXT X
140 REM **PRINT MODULE**
150 FOR X=0 TO E
160 FOR Y=0 TO 4
170 PAUSE B(X,Y)
180 NEXT Y: NEXT X
190 REM **SUM OF AY, AX, AXY, AX-2, AY-2
200 FOR X=0 TO E-1
210 L = B(X,2) * B(X,1) + L
220 M = B(X,2) * B(X,0) + M
230 N = B(X,2) * B(X,0) * B(X,1) + N
240 P = B(X,2) * SQU B(X,0) + P
250 Q = B(X,2) * SQU B(X,1) + Q
260 NEXT X
270 REM *ROW E = TOTALS ROW *
280 R=M/B(E,2) : S=L/B(E,2) : T=P+B(E,3) - SQU
    M/B(E,2) : V=Q+B(E,4) - SQU L/B(E,2)
290 WAIT 150: PRINT "X-BAR="; R
300 PRINT "Y-BAR="; S
310 PRINT "I-XY = "; N-S * M-R * L+R * S * B(E,2)
320 PRINT "I-YY="; T
330 PRINT "I-XX="; V
340 PRINT "RHO-Y="; √(T/B(E,2))
350 PRINT "RHO-X="; √(V/B(E,2))
360 WAIT: INPUT "REREAD ANS. (Y/N)", Z$
370 IF Z$ = "Y" THEN 290
380 END
```

Change to RUN Mode

Key in: **RUN**

ENTER ELEMENTS = 3

DATA =	.125	.375	4.23	2.34	2.31
	.946	1.685	5.34	2.54	2.55
	2.32	2.462	6.32	2.85	2.81

Answers:

X–BAR = 1.39692240
Y–BAR = 1.52228131
I–XY = 11.01793397
I–YY = 20.40263064
I–XX = 20.09178455
RHO–Y = 1.133133646
RHO–X = 1.124468535

ELECTRONICS

A. Parallel Resistance

The equivalent resistance of two parallel resistors is determined by the equation:

$$RT = \frac{R_1 R_2}{R_1 + R_2}$$

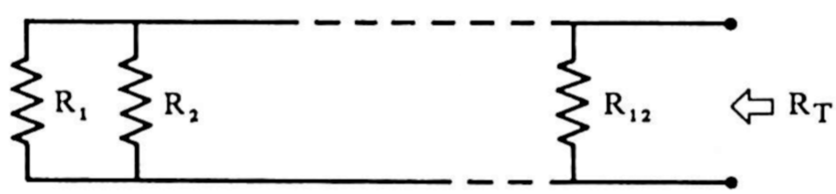


Figure 4.4 Resistors in Parallel

For more than two resistors in parallel RT is:

$$RT = \frac{1}{1/R_1 + 1/R_2 + \cdots\cdots\cdots + 1/R_n}$$

The program calculates the equivalent resistance of up to 12 resistors in parallel. The first formula is used to calculate the equivalent resistance for R_1 and R_2. This amount RT is used as one element with the next resistance R_3 to calculate a new RT. This process is continued R_{n-1} times to calculate the equivalent resistance of R_n resistors.

Line 10 and Line 40 limit the number of resistors to 12 which is large enough for most practical design problems. To increase this limit, increase the size of the **DIM** statement by 2 on Line 10 and the **FOR . . . NEXT** statement by 2 on Line 40 for each additional parallel resistor.

The **FOR . . . NEXT** loop in actual usage may never reach the upper limit of the counter. Running through the loop when N is 3, D(3) is R3 and B is 4. Therefore D(4) equals the next RT. Looping continues with all odd numbers of N being values of R and all even numbers being successive values of RT up to a maximum of 12 R's.

Line 60 controls the program and prints the value of RT when **ENTER** is pressed without a value. The expression D(N) = 0 acts as a "flag".

Program:

```
10  "A": CLEAR: DIM D(22): PAUSE "PARALLEL RESIST"
20  INPUT "R1="; D(0)
30  PRINT "R1="; D(0)
40  FOR N=1 TO 21
50  INPUT "NEXT R="; D(N)
60  IF D(N)=0 PRINT USING "########.##"; "RT="; D(B): END
70  B=N+1
80  D(B)=D(N-1) * D(N)/(D(N-1) + D(N))
90  PRINT "R"; (B+2)/2;" = "; D(N)
```

```
100  N=B
110  NEXT N
120  PRINT "RT="; D(B): USING: END
```

Determine the equivalent resistances for these 3 circuits:

a. $R_1 = 8200000$ $R_2 = 1500000$ $R_3 = 9600000$

b. $R_1 = 100$ $R_2 = 200$ $R_3 = 300$ $R_4 = 400$ $R_5 = 500$ $R_6 = 600$

c. $R_1 = 0.18$ $R_2 = 1.2$ $R_3 = 0.68$ $R_4 = 2.2$

Change to the RUN Mode.

Key in: **DEF A**

a. 82 **E** 5 **ENTER ENTER**
   15 **E** 5 **ENTER ENTER**
   96 **E** 5 **ENTER ENTER ENTER**

Answer: 1120091.06

a. *Answer:* 40.81

b. *Answer:* 0.12

B. Series Reactance and Impedance

Reactance is referred to as a "frequency-sensitive" ohm. For the capacitor it is called capacitive reactance and has the unit X_C in ohms. For inductors it is called inductive reactance and has the unit X_L in ohms. Capacitive reactance is inversely proportional to frequency while inductive reactance is directly proportional to frequency.

Impedance, Z, is the opposition to current flow caused by reactive and resistive components in series, parallel or series-parallel combinations. The program solves for Z and its phase angle for series RC, RL and RCL circuits. The type of circuit is identified on the first prompt by the program.

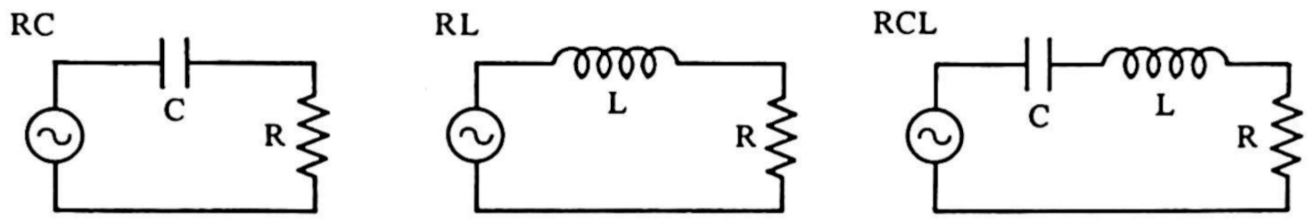


Figure 4.5 Series RC, RL, RCL Circuits

The formulas for the three types of circuits are:

RC: $Z = \sqrt{R^2 + (1/2\pi fC)^2}$ $\qquad \theta = \tan^{-1}\left[\dfrac{1/2\pi fC}{R}\right]$

RL: $Z = \sqrt{R^2 + (2\pi fL)^2}$ $\qquad \theta = \tan^{-1}\left[2\pi fL/R\right]$

RLC: $Z = \sqrt{R^2 + [(2\pi fL)^2 - (1/2\pi fC)]}$

$$\theta = \tan^{-1}\left[((2\pi fL) - (1/2\pi fC))/R\right]$$

Program:

```
10  CLEAR: U$="##.###^": INPUT "RC=1, RL=2, RCL=3"; A
15  INPUT "R="; R
20  IF A=1 OR A=3 INPUT "C="; C
30  IF A=2 OR A=3 INPUT "L="; L
60  INPUT "F="; F
70  IF A=1 PAUSE "SERIES–RC": GOTO 100
80  IF A=2 PAUSE "SERIES–RL": GOTO 150
90  IF A=3 PAUSE "SERIES–RCL": GOTO 200
100 Z=√ (SQU R + ((1/(2 * π * F * C)) ^ 2)): P=ATN ((1/(2 * π
    * F * C))/R)
110 PRINT USING U$; "R="; R
120 PRINT "C="; C
130 PRINT "F="; F
135 PRINT "Z="; Z
```

```
140 PRINT "PHASE"; P: END
150 Z=√ ( SQU R + SQU(2 * π * F * L)): P=ATN ((2 * π * F * L)/R):
    P=-P
160 PRINT USING U$; "R="; R
170 PRINT "L="; L
175 PRINT "F="; F
180 PRINT "Z="; Z
190 PRINT "PHASE"; P: END
200 Z=√ (R+(((2 * π * F * L)-(1/(2 * π * F * C)))^2)):
    P=ACS(R/Z): I=(2 * π * F * L) - (1/(2 * π * F * C))
210 IF I<0 LET P=-P
220 PRINT USING U$; "R="; R
230 PRINT "C="; C
240 PRINT "L="; L
250 PRINT "F="; F
260 PRINT "Z="; Z
270 PRINT "PHASE"; P: END
```

Obtain the Impedance and Phase angle for the following circuits:

RC:	R = 200	C = 1E–11	F = 5E7	
RL:	R = 200	L = 1	F = 60	
RCL:	R = 5	C = 1E–05	L = .025	F = 50

Change to the RUN Mode. On the first prompt, key in 1, 2, or 3 depending on the type of circuit.

Key in:

RC: 200 **ENTER** 1E–11 **ENTER** 5 E 07
ENTER

Answer: R = 2.000E 02 C = 1.000E–11
F = 5.000E 07 Z = 3.759E 02
PHASE 5.785E 01

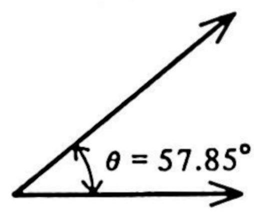


Figure 4.6 RC Phase Diagram

RL: 200 **ENTER** 1 **ENTER** 60 **ENTER**

Answer: R = 2.000E 02 L = 1.000E 00
F = 6.000E 01 Z = 4.267E 02
PHASE −6.205E 01

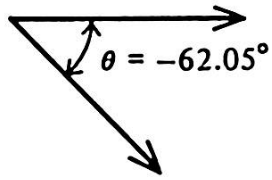


Figure 4.7 RL Phase Diagram

RCL: 5 **ENTER** 1 E−05 **ENTER** .025 **ENTER** 50 **ENTER**

Answer: R = 5.000E 00 C = 1.000E−05
L = 2.500E−02 F = 5.000E 01
Z = 3.104E 02
PHASE −8.907E 01

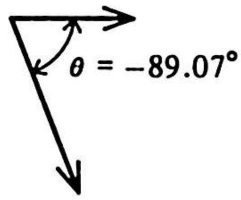


Figure 4.8 RCL Phase Diagram

C. Parallel Reactance and Impedance

The impedance of a parallel RC, RL or RCL circuit must be less than the minimum value in any one of the parallel branches. The Z_T of a parallel RC, RL or RCL circuit does not equal that of a series circuit because of the basic relatior ship of current and voltage in parallel circuits.

The program solves for the phase angle of total current with respect to resistor current. Resistor current is the reference at zero degrees. The impedance is solved by the relationship:

$$Z_T \angle 0^\circ = E/I_T \angle 0^\circ$$ which is Ohm's Law for AC circuits.

The assumed voltage for RC and RL circuits is equal to the absolute value of X_C or X_L.

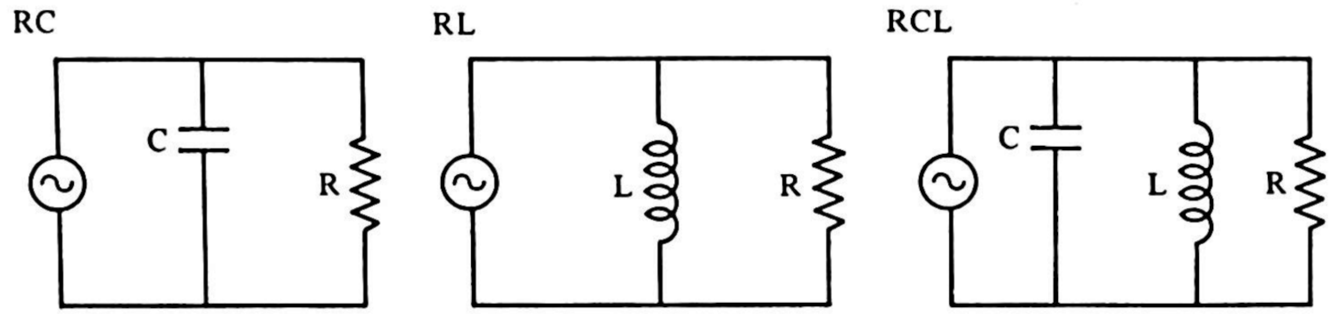


Figure 4.9 Parallel RC, RL, RCL Circuits

The formulas that apply to these three circuits are:

RC: $I_R = |X_C|/R \qquad Z = |X_C|/I_T \qquad I_T = \sqrt{(I_R^2 + 1)}$

$\theta = \tan^{-1}(1/I_R)$

RL: $I_R = |X_L|/R \qquad Z = |X_L|/I_T \qquad I_T = \sqrt{(I_R^2 + 1)}$

$\theta = \tan^{-1} - (1/I_R)$

RCL: $Z = |Z| = 1/\sqrt{1/R^2 + (1/wL - wC)^2} \qquad \theta = \tan^{-1} R(1/wL - wC)$

Program:

```
10  CLEAR: U$="##.###^": INPUT "RC=1, RL=2, RCL=3"; A
15  INPUT "R="; R
20  IF A=1 OR A=3 INPUT "C="; C
30  IF A=2 OR A=3 INPUT "L="; L
60  INPUT "F="; F
70  IF A=1 PAUSE "PARALLEL–RC": GOTO 100
80  IF A=2 PAUSE "PARALLEL–RL": GOTO 150
90  IF A=3 PAUSE "PARALLEL–RCL": GOTO 200
100 I=(1/(2*π*F*C))/R: G=√(SQU I+1): Z=(1/(2*π*F*C))/
    G: P=ATN (1/I)
110 PRINT USING U$; "R="; R
120 PRINT "C="; C
```

```
130 PRINT "F="; F
140 PRINT "Z="; Z
145 PRINT "PHASE"; P: END
150 I=2 * π * F * L/R: G=√(SQU I+1): Z=2 * π * F * L/G:
    P=ATN(-1/I)
160 PRINT USING U$; "R="; R
170 PRINT "L="; L
180 PRINT "F="; F
190 PRINT "Z="; Z
195 PRINT "PHASE"; P: END
200 Z=1/(√((1/(SQU R))+(((1/(2*π*F*L))-(2*π*F*C))^2)))
210 P=ATN(R*((1/(2*π*F*L))-(2*π*F*C))): I=1/((2*π*F*L)
    -(2 * π * F * C)): IF I<0 LET P=-P
220 PRINT USING U$; "R="; R
230 PRINT "C="; C
240 PRINT "L="; L
250 PRINT "F="; F
260 PRINT "Z="; Z
270 PRINT "PHASE"; P: END
```

Obtain the Impedance and Phase angle for the following circuits:

RC:	R = 15	C = 8E–04	F = 10	
RL:	R = 20	L = .318	F = 10	
RCL:	R = 8	C = 5E–07	L = .04	F = 60

Change to the RUN Mode. Key in 1, 2, or 3 depending on the type of circuit.

Key in:

RC: 15 **ENTER** 8E–04 **ENTER** 10 **ENTER**

Answer: R = 1.500E 01 C = 8.000E–04
F = 1.000E 01 Z = 1.197E 01
PHASE 3.701E 01

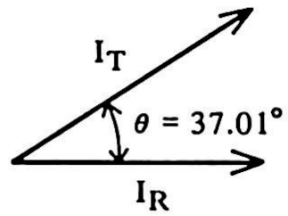


Figure 4.10 RC Phase Diagram

RL: 20 **ENTER** .318 **ENTER** 10 **ENTER**

Answer: R = 2.000E 01 L = 3.18E–01
F = 1.000E 01 Z = 1.413E 01
PHASE –4.502E 01

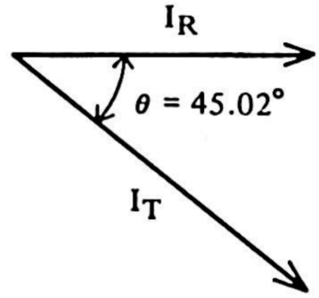


Figure 4.11 RL Phase Diagram

RCL: 8 **ENTER** 5E–07 **ENTER** .04 **ENTER** 60 **ENTER**

Answer: R = 8.000E 00 C = 5.000E–07
L = 4.000E–02 F = 6.000E 01
Z = 7.071E 00
PHASE 2.787E 01

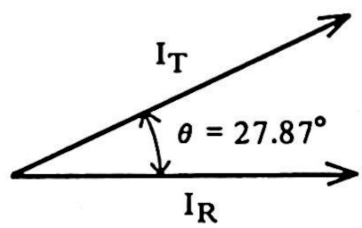


Figure 4.12 RCL Phase Diagram

D. Multiple Feedback Bandpass Filter

In the multiple feedback filter illustrated the op-amp is connected in the inverting mode and:

$$Q > \sqrt{G_0/2}$$

The formulas for the resistances are:

$$R_1 = Q/(2\pi f G_0 C) \qquad R_2 = Q/(2\pi f C(2Q^2 - G_0)) \qquad R_3 = 2R_1 G_0$$

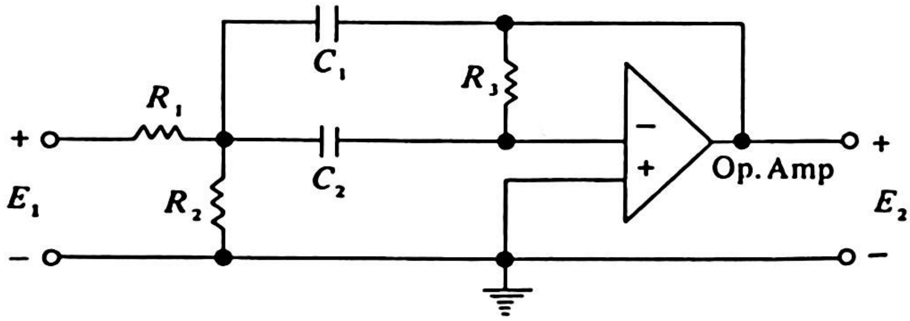


Figure 4.13 Multiple Feedback Filter

The center frequency can easily be shifted. The center frequency gain and bandwidth remain constant with the new value of R because:

$$R' = R_2 (f/f')^2$$

Program:

```
10   "A": CLEAR: PAUSE "BANDPASS FILTER": INPUT
     "CENTER FREQ"; F: INPUT "GAIN"; G: INPUT "Q="; Q:
     INPUT "C="; C
20   X=Q/(2 * π * F * G * C)
30   Z=2 * X * G
40   Y=Q/((2 * π * F * C) * ((2 * SQU Q) - 1.32))
50   PRINT USING "##.##^"; "CENT.F."; F
60   PRINT "GAIN="; G
70   PRINT "Q="; Q
80   PRINT "C="; C
90   PRINT "R1="; X
100  PRINT "R2="; Y
110  PRINT "R3="; Z
120  INPUT "NEW CENTER FREQ"; H
130  Y=Y * SQU (F/H)
140  PRINT "R2="; Y: END
```

Design an 800 Hz bandpass filter using the circuit given. The center frequency gain is 1.25 and Q is 3.8. The capacitors are selected from standard values. Using .022 Mf determine R_1, R_2 and R_3. Shift the center frequency to 600 Hz.

Change to the RUN Mode.

Key in:

800 **ENTER** 1.25 **ENTER** 3.8 **ENTER** .022 **E–6** **ENTER**

Answers:

CENT.F. 8.00E 02 GAIN = 1.25E 00 Q = 3.80E 00
C = 2.20E–08 R_1 = 2.74E 04 R_2 = 1.24E 03 R_3 = 6.87E 04

Key in: 600 **ENTER** *Answer:* R_2 = 2.21E 03

MECHANICS

A. Solar Power

The most important characteristic of a solar collector is the percent of solar radiation transmitted by the glass to the absorbing material (water). The radiation transmitted is a function of the glass's refractive index (for ordinary glass = 1.52) and the angle of incidence of the sun.

The reflectance from a plate of glass varies from the incidence angle (i). The fresnel equation calculates the reflection of a beam and the simultaneous radiation through one surface of a transparent solid.

$$\text{RHO} = 1/2\left[\frac{\sin^2(i-r_a)}{\sin^2(i+r_a)} + \frac{\tan^2(i-r_a)}{\tan^2(i+r_a)}\right]$$

r_a is the angle of refraction and is calculated by:

$$\sin r_a = \sin i/r_i\text{-coll} \qquad r_i\text{-coll} = \text{refractive index} = 1.52$$

In the program $I = i$ and $R = r_a$.

Program:

```
10 CLEAR: PAUSE "SOL * COLLECTOR"
20 REM-- INCIDENCE ANGLE IS I
30 FOR I=5 TO 90 STEP 5
40 R=ASN ( SIN I/1.52)
50 X=I-R: Y=I+R
60 H=(SQU SIN X/SQU SIN Y+SQU TAN X/SQU TAN Y)/2
70 PRINT "RHO=", USING "####.##"; 100 * H
80 NEXT I
90 END
```

Change to the RUN Mode and run the program. Each time ENTER is pressed, a new value for the reflectance will appear.
These are plotted in Figure 4.14

I:	5	10	15	20	25	30	35	40	45	50
RHO:	4.25	4.25	4.26	4.28	4.32	4.41	4.57	4.84	5.30	6.05

I:	55	60	65	70	75	80	85	90
RHO:	7.28	9.24	12.40	17.47	25.67	39.09	61.49	99.99

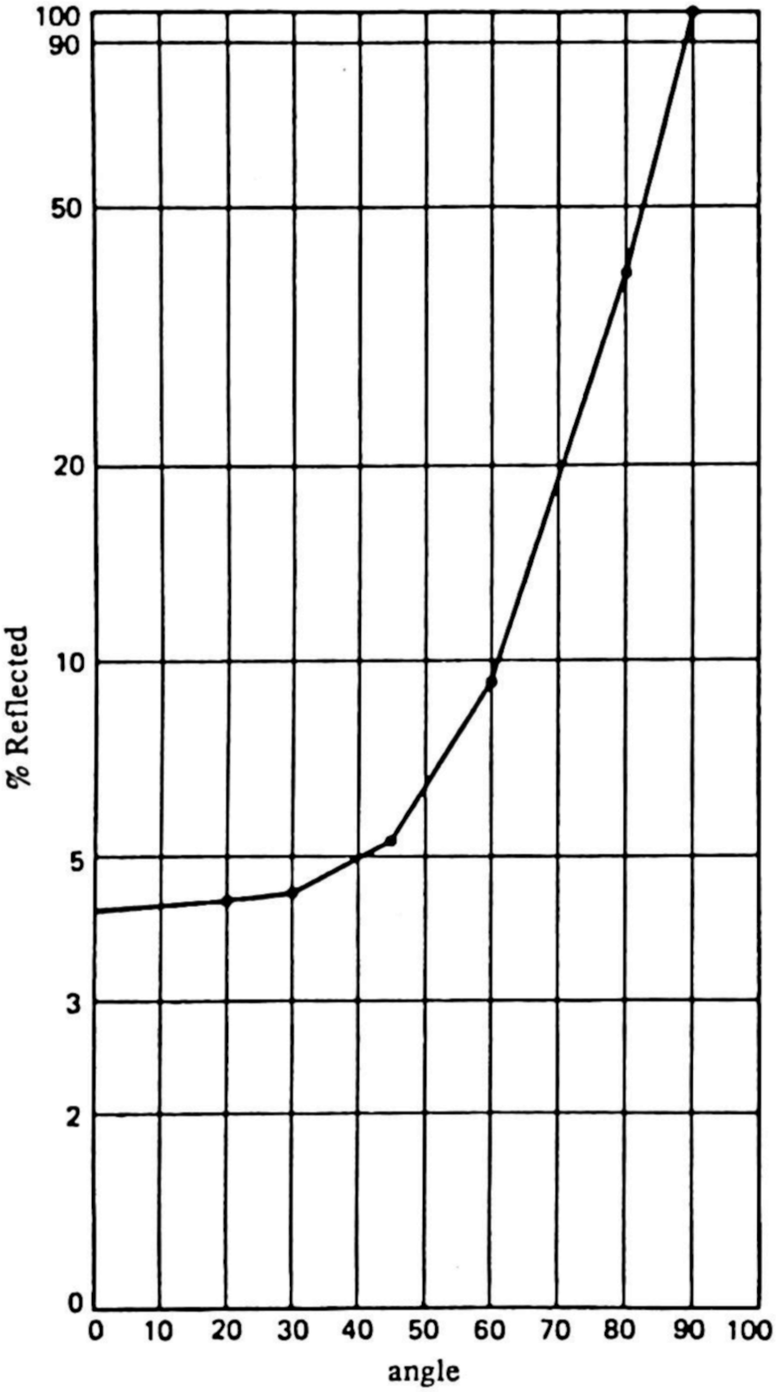


Figure 4.14 % Reflected

B. Head

Head is the pressure consumed (or "lost") in forcing a fluid along against the resistances caused by the conduits through which the fluid flows. The equation for the head lost by friction in a water pipe is:

$$H = \frac{.2083 * (100/C)^{\wedge}1.85}{D^{\wedge}4.8655}$$

where H = loss of head due to friction

C = factor for type of pipe

D = inside diameter of pipe

The program assumes new pipe and considers elbows and valves. The factor for valves is based on the assumption that the number of Gate valves equals the number Globe valves. If the system primarily contains Globe valves, double the actual number of valves.

Other assumptions are:

1. Barometric pressure equals sea level.
2. The liquid in the pipe is water.
3. Water temperature is between 70°F and 200°F

Flow may be input into the program as either gallons per minute or in terms of the velocity as feet per second. On lines 40 to 70 the **WAIT** 50 command causes the names of the various types of pipe to flash on the screen at short intervals. **WAIT** on the following line by itself removes the previous **WAIT** command. The use of **ON** in Line 100 eliminates four **IF...THEN** statements and indicates multiple use of a line. On line 100 the **BEEP** reaffirms the number selected.

Program:

```
10   "F": CLEAR: INPUT "ENTER PIPE I.D."; D
20   INPUT "ENTER G.P.M. "; Q
30   IF Q = 0 THEN INPUT "VELOCITY (FPS)?"; V:
     Q=2.448 * SQU D * V
40   WAIT 50: PRINT "1 = STEEL"
50   PRINT "2 = BRASS/COPPER"
60   PRINT "3 = CONCRETE"
70   PRINT "4 = PVC/POLY"
```

```
80   WAIT: INPUT "ENTER 1–4"; Z
90   IF Z>4 OR Z<1 THEN 40
100  BEEP Z: ON Z GOTO 110, 120, 110, 130
110  C = 100:  GOTO 140
120  C = 130:  GOTO 140
130  C = 140:  GOTO 140
140  F = .2083 * (100/C) ^ 1.85 * Q ^ 1.85/ D ^ 4.8655:
     F = F/100
150  INPUT "NO. OF 45 FENDS"; B
160  INPUT "NO. OF 90 BENDS"; N
170  INPUT "NO. OF VALVES"; V
180  W = V * (53.75 * 1 * D ^ 1.25)
190  B = B + 2 * N
200  E = 53.75 * .33 * ( D/12) ^ 1.25
210  E = B * D * E
220  REM – EQUIVALENT LENGTH IS LENGTH PLUS ELBOWS &
     VALVES ––
230  INPUT "ENTER LENGTH"; L: L = L + E + W
240  PRINT "H = "; F * L : END
```

Calculate the minimum head required for water to flow at a rate of 400 gallons per minute through 100 feet of brass pipe with an internal diameter of 8 inches, four 45° bends, two 90° bends and four valves.

Change to RUN Mode.

Key in:

8 **ENTER** 400 **ENTER** 2 **ENTER** 4 **ENTER** 2 **ENTER** 4 **ENTER** 100 **ENTER**

Answer: H = 12.40538765 feet

GAMES

A. Memory Taster

Running the program will cause a number to appear briefly on the screen. Key in the number after you see it. The computer will give you a rating corresponding to the number of times you answer correctly.

The program determines the length of time the number stays on the screen depending on whether the user selects fast (F), medium (M) or slow (S). The length of time is a function of the **WAIT** command on Line 80. Two beeps occur when a new number is placed on the screen and three if the answer is correct. The number is determined by the use of **RANDOM** and **RND**.

Program:

```
10   "A": RANDOM: K = 0: L = 0
20   INPUT "SPEED (F, M, S)?"; B$
30   IF B$ = "F" THEN LET X = 10
40   IF B$ = "M" THEN LET X = 50
50   IF B$ = "S" THEN LET X = 100
60   FOR I = 1 TO 10
70   N = RND (9999)
80   BEEP 2: WAIT X: PRINT N
90   INPUT "ANSWER ="; T
100  IF T = N BEEP 3: L = L + 1: GOTO 130
110  K = K + 1
120  IF K=4 THEN WAIT 80: PRINT "TRY SLOWER SPEED": END
130  NEXT I
135  WAIT 80: PRINT "# CORRECT=", L
140  IF L = 10 THEN PRINT "THE GREATEST": END
150  IF L =  9 THEN PRINT "VERY GOOD": END
160  IF L =  8 THEN PRINT "GOOD": END
170  IF L =  7 THEN PRINT "AVERAGE": END
```

```
180 IF L =  6 THEN PRINT "RUNNING DOWN": END
190 IF L =  5 THEN PRINT "NOT SO HOT": END
```

Change to RUN Mode. Select the speed and play the game.

B. NUMBER CHOICE

The random generator selects a number for you between 0 and 999. Each time you guess a number, the computer will indicate how far you are from the computer generated number.

SIGNAL	DISTANCE FROM THE NUMBER
FAR OFF	Over 200 away from the answer
DISTANT	Between 100 and 200 from the answer
CLOSER	Between 50 and 100 from the answer
VERY CLOSE	Between 25 and 50 from the answer
HOT!!	Between 10 and 25 from the answer
ON TARGET	Less than 10 separates you from the answer
ONE????	Too high or too low by only one

```
10  CLEAR: RANDOM: N= RND (999): L=0
20  INPUT "GUESS=";G
25  L=L+1: WAIT 100
30  IF G>N+200 OR G<N-200 PRINT "FAR OFF": GOTO 20
40  IF G>N+100 OR G<N-100 PRINT "DISTANT": GOTO 20
50  IF G>N+50  OR G<N-50  PRINT "CLOSER": GOTO 20
60  IF G>N+25  OR G<N-25  PRINT "VERY CLOSE": GOTO 20
70  IF G>N+10  OR G<N-10  PRINT "HOT!!": GOTO 20
80  IF G>N+ 1  OR G<N- 1  PRINT "ON TARGET": GOTO 20
90  IF G=N-1 OR G=N+1 PRINT "ONE????": GOTO 20
100 IF G=N  BEEP 3: PRINT "BULLSEYE"
110 PRINT "NO. OF TRIES=";L
120 END
```

APPENDIX A

Table I

Metric Length Relationships

	μ	mm	cm	deci-meter	METER	deka-meter	hm	km	mega-meter
1 micron =	1	.001	10^{-4}	10^{-5}	10^{-6}	10^{-7}	10^{-8}	10^{-9}	10^{-12}
1 millimeter =	10^3	1	.1	.01	.001	10^{-4}	10^{-5}	10^{-6}	10^{-9}
1 centimeter =	10^4	10	1	.1	.01	.001	10^{-4}	10^{-5}	10^{-8}
1 decimeter =	10^5	100	10	1	.1	.01	.001	10^{-4}	10^{-7}
1 METER =	10^6	1000	100	10	1	.1	.01	10^{-3}	10^{-6}
1 dekameter =	10^7	10^4	1000	100	10	1	.1	.01	10^{-5}
1 hectometer =	10^8	10^5	10^4	10^3	100	10	1	.1	10^{-4}
1 kilometer =	10^9	10^6	10^5	10^4	10^3	100	10	1	10^{-3}
1 megameter =	10^{12}	10^9	10^8	10^7	10^6	10^5	10^4	10^3	1

Table II

CONVERSION FACTORS

Unit	Multiply By	To Get
Ampere-hours	3600	Coulombs
Ampere-hours	0.037307	Faradays
Atmospheres	76	Cm. of Hg (0°C)
Atmospheres	14.6960	Pounds/square inch
B.T.U.	252	Cal., gram
B.T.U.	1.05508×10^{10}	Ergs
B.T.U.	778.184	Foot-pounds
B.T.U.	1055	Joules
B.T.U./hour	0.000393022	Horsepower
B.T.U./hour	0.000293018	Kilowatts (int.)
Calories, gram	0.00396832	B.T.U.
Calories, gram	4.18689×10^7	Ergs
Calories, gram	3.08808	Foot-pounds
Calories, gram	4.18689	Joules
Calories, gram/hr.	1.55964×10^{-6}	Horsepower

Calories, gram/hr.	1.16279×10^{-6}	Kilowatts
Cm. of Hg (0°C)	0.0131579	Atmospheres
Coulombs	0.000277778	Ampere-hours
Coulombs	1.0365×10^{-5}	Faradays
Dynes	2.248089×10^{-6}	Pounds
Dynes/square cm.	1×10^{-6}	Bars
Ergs	9.47798×10^{-11}	B.T.U.
Ergs	2.38841×10^{-8}	Cal., gram
Ergs	7.3756×10^{-8}	Foot/pounds
Ergs	1×10^{-7}	Joules
Ergs/sec.	1.34102×10^{-10}	Horsepower
Faradays	26.8	Ampere-hours
Faradays	96.480	Coulombs
Feet of H_2O (60°F)	0.029469	Atmospheres
Feet of H_2O (60°F)	62.364	Pounds/square foot
Feet/hour	1.64468×10^{-5}	Knots
Feet/second	0.681818	Miles/hour
Foot-pounds	0.00128504	B.T.U.
Foot-pounds	0.323826	Cal., gram
Foot-pounds	1.35582×10^{7}	Ergs
Foot-pounds	1.35582	Joules
Foot-pounds	3.76617×10^{-7}	Kilowatt-hr.
Foot-pounds/minute	3.0303×10^{-5}	Horsepower
Gram-cm.	9.29472×10^{-8}	B.T.U.
Gram-cm.	2.34223×10^{-5}	Cal., gram
Gram-cm.	980.665	Ergs
Gram-cm.	7.2330×10^{-5}	Foot-pounds
Horsepower	2544.39	B.T.U./hour
Horsepower	10686.3	Cal., gram/min.
Horsepower	7.45702×10^{9}	Ergs/sec.
Horsepower	550	Foot-pounds/sec.
Horsepower	0.745702	Kilowatts
Inches of Hg (32°F)	0.033421	Atmospheres
Inches of Hg (32°F)	33864	Dynes/square cm.
Inches of Hg (32°F)	70.726	Pounds/square foot

Joules	0.000947798	B.T.U.
Joules	0.238841	Cal., gram
Joules	1×10^7	Dynes-cm.
Joules	1×10^7	Ergs
Joules	0.73756	Foot-pounds
Joules	10197.16	Gram-cm.
Kilowatts	3412.08	B.T.U./hour
Kilowatts	859,828	Cal., gram/hour
Kilowatts	1×10^{10}	Ergs/second
Kilowatts	2.65522×10^6	Foot-pounds/hour
Kilowatts	1.34102	Horsepower
Kilowatts	3.67098×10^5	Kg.-meters/hour
Kilowatt-hours	3.6×10^6	Joules
Knots	1.8532486	Kilometers/hour
Knots	1	Miles (Naut.)/hour
Knots	1.15155	Miles/hour
Lamberts	295.719	Candles/square foot
Lamberts	929.034	Lumens/square foot
Light years	9.45994×10^{12}	Kilometers
Light years	5.87812×10^{12}	Miles
Meters of Hg (0°C)	1.31579	Atmospheres
Meters of Hg (0°C)	44.650	Feet of H_2O (60°F)
Meters of Hg (0°C)	19.3368	Pounds/square inch
Miles/hour	1.46667	Feet/sec.
Million gal./day	1.54723	Cu. ft./sec.
Ounces/sq. in.	0.0106042	Feet of Hg (32°F)
Ounces/sq. in.	0.144314	Feet of H_2O (60°F)
Pounds/square foot	0.000472543	Atmospheres
Pounds/square foot	0.000478803	Bars
Pounds/square foot	0.0359131	Cm. of Hg (0°C)
Pounds/square foot	478.803	Dynes/sq. cm.
Pounds/square foot	0.00117825	Ft. of Hg (32°F)
Pounds/square foot	0.016035	Ft. of H_2O (60°F)
Pounds/square foot	0.488241	Gram/sq. cm.
Watts	3.413	B.T.U./hour

Watts	860	Cal., gram/hour
Watts	1×10^7	Ergs/second
Watts	44.25	Foot-pounds/minute
Watts	0.001341	Horsepower
Watt-hours	2655.75	Foot-pounds

Table III
FUNDAMENTAL CONSTANTS

Constant	Symbol	Value: Rounded	Value: "Exact"*	Unit
Speed of Light	c	3.00×10^8	2.99792458	m/s
Gravitational Constant	G	6.67×10^{-11}	6.6720	$m^3/s^2 \cdot kg$
Electric Charge	e	1.60×10^{-19}	1.6021892	C
Electron Mass	m_e	9.11×10^{-31}	9.109534	kg
Proton Mass	m_p	1.67×10^{-27}	1.6726485	kg
Neutron Mass	m_n	1.68×10^{-27}	1.6749543	kg
Planck Constant	h	6.63×10^{-34}	6.626176	$J \cdot s$
Avogadro's Number	N_A	6.02×10^{23}	6.022045	/mol
Gas Constant	R	8.31	8.31441	$J/mol \cdot K$
Boltzmann Constant	k	1.38×10^{-23}	1.380662	J/K
Ideal Gas Volume at STP	V_m	2.24×10^{-2}	2.241383	m^3/mol
Faraday Constant	F	9.65×10^4	9.648456	C/mol
Stefan-Boltzmann Constant	σ	5.67×10^{-8}	5.67032	$W/m^2 \cdot K^4$
Bohr Radius	a_0	5.29×10^{-11}	5.2917706	m

* Same power of ten as rounded value.

Table IV

WEIGHTS AND MEASURES

Length

1 Nautical Mile = 1.15151 miles
1 Mile = 1760 yards = 5280 feet
1 Yard = 3 feet = 36 inches
1 Foot = 12 inches

Area

1 Square Mile = 640 acres
1 Acre = 4840 sq. yards = 43,560 sq. feet
1 Square Yard = 9 sq. feet
1 Square Feet = 144 sq. inches

Volume

1 Cubic Yard = 27 cubic feet
1 Cubic Feet = 1728 cubic inches
1 Quart = 2 pints
1 U.S. Gallon = 8 U.S. Pints
1 U.S. Bushel = 64 U.S. Pints

Weight

1 Long Ton = 2240 pounds
1 Short Ton = 2000 pounds
1 Pound = 16 ounces
1 Ounce = 437.5 grains

Time

1 Year = 365 days (366 in a leap year)
1 Month = 30 days (or 31, or 28 & 29 for February)
1 Week = 7 days
1 Hour = 60 minutes = 3600 seconds
1 Minute = 60 seconds

Table V

Greek Letters and Roman Numerals

The Greek Alphabet

Name	Upper case	Lower Case
Alpha	A	α
Beta	B	β
Gamma	Γ	γ
Delta	Δ	δ
Epsilon	E	ϵ
Zeta	Z	ζ
Eta	H	η
Theta	Θ	θ
Iota	I	ι
Kappa	K	κ
Lambda	Λ	λ
Mu	M	μ
Nu	N	ν
Xi	Ξ	ξ
Omicron	O	o
Pi	Π	π
Rho	P	ρ
Sigma	Σ	σ
Tau	T	τ
Upsilon	Υ	υ
Phi	Φ	ϕ
Chi	X	χ
Psi	Ψ	ψ
Omega	Ω	ω

ROMAN AND ARABIC NUMERALS

I	1	XX	20
II	2	XXX	30
III	3	XL	40
IV	4	L	50
V	5	LX	60
VI	6	LXX	70
VII	7	LXXX	80
VIII	8	XC	90
IX	9	C	100
X	10	CC	200
XI	11	CCC	300
XII	12	CCCC	400
XIII	13	D	500
XIV	14	DC	600
XV	15	DCC	700
XVI	16	DCCC	800
XVII	17	CM	900
XVIII	18	M	1000
XIX	19		

Table VI

	IA	IIA	IIIB	IVB	VB	VIB	VIIB	VIII			IB	IIB	IIIA	IVA	VA	VIA	VIIA	0
1	1.0080 H 1																	4.003 He 2
2	6.939 Li 3	9.012 Be 4											10.81 B 5	12.011 C 6	14.007 N 7	15.999 O 8	18.998 F 9	20.183 Ne 10
3	22.991 Na 11	24.31 Mg 12											26.98 Al 13	28.09 Si 14	30.974 P 15	32.06 S 16	35.453 Cl 17	39.948 Ar 18
4	39.102 K 19	40.08 Ca 20	44.96 Sc 21	47.90 Ti 22	50.94 V 23	52.00 Cr 24	54.94 Mn 25	55.85 Fe 26	58.93 Co 27	58.71 Ni 28	63.54 Cu 29	65.37 Zn 30	69.72 Ga 31	72.59 Ge 32	74.92 As 33	78.96 Se 34	79.909 Br 35	83.80 Kr 36
5	85.47 Rb 37	87.62 Sr 38	88.91 Y 39	91.22 Zr 40	92.91 Nb 41	95.94 Mo 42	Tc 43	101.1 Ru 44	102.91 Rh 45	106.4 Pd 46	107.87 Ag 47	112.40 Cd 48	114.82 In 49	118.69 Sn 50	121.75 Sb 51	127.60 Te 52	126.90 I 53	131.30 Xe 54
6	132.91 Cs 55	137.34 Ba 56	138.92 La 57	178.49 Hf 72	180.95 Ta 73	183.85 W 74	186.2 Re 75	190.2 Os 76	192.2 Ir 77	195.09 Pt 78	196.97 Au 79	200.59 Hg 80	204.37 Tl 81	207.19 Pb 82	208.98 Bi 83	Po 84	At 85	222 Rn 86
7	Fr 87	Ra 88	Ac 89															

Lanthonide Series	140.12 Ce 58	140.91 Pr 59	144.24 Nd 60	Pm 61	150.35 Sm 62	151.96 Eu 63	157.25 Gd 64	158.92 Tb 65	162.50 Dy 66	164.94 Ho 67	167.26 Er 68	168.93 Tm 69	173.04 Yb 70	174.97 Lu 71
Actinide Series	232.04 Th 90	231 Pa 91	238.03 U 92	237 Np 93	Pu 94	Am 95	Cm 96	Bk 97	Cf 98	Es 99	Fm 100	Md 101	No 102	Lw 103

A PERIODIC CHART OF THE ELEMENTS (BASED ON 12C)

Table VII

Hexadecimal addition and multiplication

ADDITION TABLE

+	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10
2	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11
3	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12
4	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13
5	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14
6	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15
7	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16
8	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17
9	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18
A	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19
B	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A
C	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B
D	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C
E	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D
F	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E

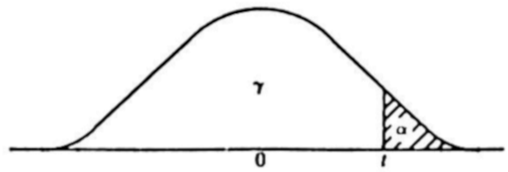
MULTIPLICATION TABLE

X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	0	2	4	6	8	A	C	E	10	12	14	16	18	1A	1C	1E
3	0	3	6	9	C	F	12	15	18	1B	1E	21	24	27	2A	2D
4	0	4	8	C	10	14	18	1C	20	24	28	2C	30	34	38	3C
5	0	5	A	F	14	19	1E	23	28	2D	32	37	3C	41	46	4B
6	0	6	C	12	18	1E	24	2A	30	36	3C	42	48	4E	54	5A
7	0	7	E	15	1C	23	2A	31	38	3F	46	4D	54	5B	62	69
8	0	8	10	18	20	28	30	38	40	48	50	58	60	68	70	78
9	0	9	12	1B	24	2D	36	3F	48	51	5A	63	6C	75	7E	87
A	0	A	14	1E	28	32	3C	46	50	5A	64	6E	78	82	8C	96
B	0	B	16	21	2C	37	42	4D	58	63	6E	79	84	8F	9A	A5
C	0	C	18	24	30	3C	48	54	60	6C	78	84	90	9C	A8	B4
D	0	D	1A	27	34	41	4E	5B	68	75	82	8F	9C	A9	B6	C3
E	0	E	1C	2A	38	46	54	62	70	7E	8C	9A	A8	B6	C4	D2
F	0	F	1E	2D	3C	4B	5A	69	78	87	96	A5	B4	C3	D2	E1

Table VIII Probability Levels for the Student's t-distribution

γ = area below t
  = confidence coefficient

$\alpha = \frac{1}{2}(1-\gamma)$
  = area above t
  = significance level of one-sided test
  one-sided test



df	One Sample Size	P =	.55	.60	.70	.75	.80	.90	.95	.975	.99	.995	.9995
		γ / α	.45	.40	.30	.25	.20	.10	.05	.025	.01	.005	.0005
1	2		.158	.325	.727	1.000	1.376	3.08	6.31	12.71	31.82	63.66	636.61
2	3		.142	.289	.617	.816	1.061	1.89	2.92	4.30	6.96	9.92	31.60
3	4		.137	.277	.584	.765	.978	1.64	2.35	3.18	4.54	5.84	12.94
4	5		.134	.271	.569	.741	.941	1.53	2.13	2.78	3.75	4.60	8.61
5	6		.132	.267	.559	.727	.920	1.48	2.02	2.57	3.36	4.03	6.86
6	7		.131	.265	.553	.718	.906	1.44	1.94	2.45	3.14	3.71	5.96
7	8		.130	.263	.549	.711	.896	1.42	1.90	2.36	3.00	3.50	5.41
8	9		.130	.262	.546	.706	.889	1.40	1.86	2.31	2.90	3.36	5.04
9	10		.129	.261	.543	.703	.883	1.38	1.83	2.26	2.82	3.25	4.78
10	11		.129	.260	.542	.700	.879	1.37	1.81	2.23	2.76	3.17	4.59
11	12		.129	.260	.540	.697	.876	1.36	1.80	2.20	2.72	3.11	4.44
12	13		.128	.259	.539	.695	.873	1.36	1.78	2.18	2.68	3.06	4.32
13	14		.128	.259	.538	.694	.870	1.35	1.77	2.16	2.65	3.01	4.22
14	15		.128	.258	.537	.692	.868	1.34	1.76	2.14	2.62	2.98	4.14
15	16		.128	.258	.536	.691	.866	1.34	1.75	2.13	2.60	2.95	4.07
16	17		.128	.258	.535	.690	.865	1.34	1.75	2.12	2.58	2.92	4.02
17	18		.128	.257	.534	.689	.863	1.33	1.74	2.11	2.57	2.90	3.97
18	19		.127	.257	.534	.688	.862	1.33	1.73	2.10	2.55	2.88	3.92
19	20		.127	.257	.533	.688	.861	1.33	1.73	2.09	2.54	2.86	3.88
20	21		.127	.257	.533	.687	.860	1.32	1.72	2.09	2.53	2.84	3.90
21	22		.127	.257	.532	.686	.859	1.32	1.72	2.08	2.52	2.83	3.82
22	23		.127	.256	.532	.686	.858	1.32	1.72	2.07	2.51	2.82	3.79
23	24		.127	.256	.532	.685	.858	1.32	1.71	2.07	2.50	2.81	3.77
24	25		.127	.256	.531	.685	.857	1.32	1.71	2.06	2.49	2.80	3.75
25	26		.127	.256	.531	.684	.856	1.32	1.71	2.06	2.48	2.79	3.73
26	27		.127	.256	.531	.684	.856	1.32	1.71	2.06	2.48	2.78	3.71
27	28		.127	.256	.531	.684	.855	1.31	1.70	2.05	2.47	2.77	3.69
28	29		.127	.256	.530	.683	.855	1.31	1.70	2.05	2.47	2.76	3.67
29	30		.127	.256	.530	.683	.854	1.31	1.70	2.04	2.46	2.76	3.66
30	31		.127	.256	.530	.683	.854	1.31	1.70	2.04	2.46	2.75	3.65
40	41		.126	.255	.529	.681	.851	1.30	1.68	2.02	2.42	2.70	3.55
60	61		.126	.254	.527	.679	.848	1.30	1.67	2.00	2.39	2.66	3.46
120	121		.126	.254	.526	.677	.845	1.29	1.66	1.98	2.36	2.62	3.37
			.126	.253	.524	.674	.842	1.28	1.65	1.96	2.33	2.58	3.30
df	One Sample Size	P = γ	.10	.20	.40	.50	.60	.80	.90	.95	.98	.99	.999
		$\alpha+\alpha$	.90	.80	.60	.50	.40	.20	.10	.05	.02	.01	.001

γ = area between $-t$ and $+t$
  = confidence coefficient

$\alpha + \alpha = 1 - \gamma = 2\alpha$
  = area beyond $-t$ and t
  = significance level of
  two-sided test

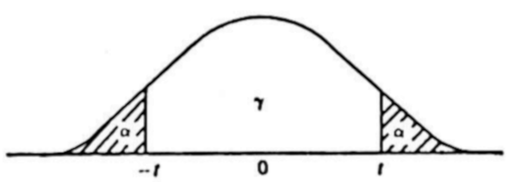
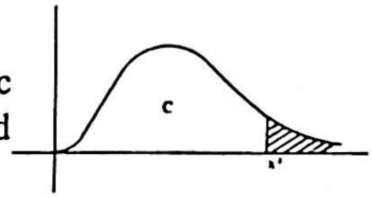


Table IX CHI-SQUARE (χ^2) DISTRIBUTION

UPPER TAIL PROBABILITIES*

WHERE $\chi^2 > C$

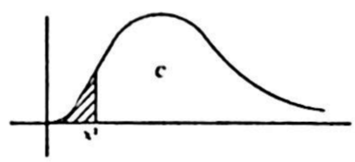
NOTE: The shape of the curve reflects the non-parametric nature of this process, i.e., the data is not based on the concept of a normal curve.



df	* .990	.980	.975	.950	.900	.750	.500	.250	.200	.100	.050	.025	.020	.010	.005	.001
1	$.0^3157$	$.0^3628$	.011	.004	.016	.102	.455	1.32	1.64	2.71	3.84	5.02	5.41	6.64	7.88	10.83
2	.020	.040	.051	.103	.211	.575	1.39	2.77	3.22	4.61	5.99	7.38	7.82	9.21	10.60	13.82
3	.115	.185	.216	.352	.584	1.21	2.37	4.11	4.64	6.25	7.82	9.35	9.84	11.35	12.84	16.27
4	.279	.429	.484	.711	1.06	1.92	3.38	5.39	5.99	7.78	9.49	11.14	11.67	13.28	14.86	18.47
5	.554	.752	.831	1.15	1.61	2.67	4.35	6.63	7.29	9.24	11.07	12.83	13.39	15.09	16.75	20.52
6	.872	1.13	1.24	1.64	2.20	3.45	5.35	7.84	8.56	10.65	12.59	14.45	15.03	16.81	18.55	22.46
7	1.24	1.56	1.69	2.17	2.83	4.25	6.35	9.04	9.80	12.02	14.07	16.01	16.62	18.48	20.28	24.32
8	1.65	2.03	2.18	2.73	3.49	5.07	7.34	10.2	11.03	13.36	15.51	17.53	18.17	20.09	21.96	26.16
9	2.09	2.53	2.70	3.33	4.16	5.90	8.34	11.4	12.24	14.68	16.92	19.02	19.68	21.67	23.59	27.88
10	2.56	3.06	3.25	3.94	4.87	6.74	9.34	12.5	13.44	15.99	18.31	20.48	21.16	23.21	25.19	29.59
11	3.05	3.61	3.82	4.58	5.58	7.58	10.34	13.7	14.63	17.28	19.68	21.92	22.62	24.73	26.76	31.26
12	3.57	4.18	4.40	5.23	6.30	8.44	11.34	14.8	15.81	18.55	21.03	23.34	24.05	26.22	28.30	32.99
13	4.11	4.77	5.01	5.89	7.04	9.30	12.34	16.0	16.99	19.81	22.36	24.74	25.47	27.69	29.82	34.53
14	4.66	5.37	5.63	6.57	7.79	10.2	13.34	17.1	18.15	21.06	23.69	26.12	26.87	27.14	31.32	36.12
15	5.23	5.99	6.26	7.26	8.55	11.0	14.34	18.2	19.31	22.31	25.00	27.49	28.26	30.59	32.80	37.70
16	5.81	6.61	6.91	7.96	9.31	11.9	15.34	19.4	20.47	23.54	26.30	28.85	29.63	32.00	34.27	39.25
17	6.41	7.26	7.56	8.67	10.09	12.8	16.34	20.5	21.62	24.77	27.59	30.19	31.00	33.41	35.72	40.79
18	7.02	7.99	8.23	9.39	10.87	13.7	17.34	21.6	22.76	25.99	28.87	31.53	32.35	34.81	37.16	42.31
19	7.63	8.57	8.91	10.12	11.65	14.6	18.34	22.7	23.90	27.20	30.14	32.85	33.69	36.19	38.58	43.82
20	8.26	9.24	9.59	10.85	12.44	15.5	19.34	23.8	25.04	28.41	31.41	34.17	35.02	37.57	40.00	45.32
21	8.90	9.92	10.3	11.59	13.24	16.3	20.34	24.9	26.17	29.62	32.67	35.48	36.34	38.93	41.40	46.80
22	9.54	10.60	11.0	12.34	14.04	17.2	21.34	26.0	27.30	30.81	33.92	36.78	37.66	40.29	42.80	48.27
23	10.20	11.29	11.7	13.09	14.85	18.1	22.34	27.1	28.43	32.01	35.17	38.08	38.97	41.64	44.18	49.72
24	10.86	11.99	12.4	13.84	15.66	19.0	23.34	28.2	29.55	33.20	36.42	39.36	40.27	42.98	45.56	51.18
25	11.52	12.70	13.1	14.61	16.47	19.9	24.34	29.3	30.68	34.38	37.65	40.65	41.57	44.31	46.93	52.62
26	12.20	13.41	13.8	15.38	17.29	20.8	25.34	30.4	31.80	35.56	38.89	41.92	42.86	45.64	48.29	54.05
27	12.88	14.13	14.6	16.15	18.11	21.7	26.34	31.5	32.91	36.74	40.11	43.19	44.14	46.96	49.64	55.48
28	13.57	14.85	15.3	16.93	18.94	22.7	27.34	32.6	34.03	37.92	41.34	44.46	45.42	48.29	50.99	56.80
29	14.26	15.57	16.0	17.70	19.77	23.6	28.34	33.7	35.14	39.09	42.56	45.72	46.69	49.58	52.34	58.30
30	14.95	16.31	16.8	18.49	20.60	24.5	29.34	34.8	36.25	40.26	43.77	46.98	47.96	50.89	53.67	59.70
40	22.2		24.4	26.5	29.1	33.7	49.3	45.6		51.8	55.76	59.34		63.69	66.77	
50											67.50	71.42		76.15	79.49	
60	37.5		40.5	43.2	46.5	52.3	59.3	67.0		74.4	79.08	83.30		88.38	91.95	
70											90.53	95.02		100.43	104.22	
80											101.88	106.63		112.33	116.32	
90											113.15	118.14		124.12	128.30	
100											124.34	129.56		135.81	140.17	
	**.010	.020	.025	.050	.100	.250	.500	.750	.800	.900	.950	.975	.980	.990	.995	.999

LOWER TAIL PROBABILITIES**

WHERE $\chi^2 < C$



APPENDIX B

Glossary of Terms

annunciators Small messages that are in the screen of the EL-5500III that the computer uses to communicate what mode it is in. **BUSY, DEG, RAD, GRAD, P,** and **E** are all used by the EL-5500III.

array A set of variables denoted by subscripting of a single variable name. Can be either numeric or string.

ASCII The set of codes used in many computers to represent characters.

BASIC A language for communicating with a computer. The letters stand for Beginners All-purpose Symbolic Instruction Code. One of the most widely used languages in existence today.

binary A system of numbers comprised of two digits: 0 and 1. See decimal, hexadecimal, and octal.

binary number A number expressed in Binary.

bit A binary digit; either a 0 or a 1.

bug A problem or error in a program that causes the program to not run properly. Almost always due to insufficient planning of a program or a typographical error in entering a program into the computer's memory.

byte A binary number that is eight binary bits or three octal digits or two hexadecimal digits long. Generally can be thought of as one character.

character Any piece of information that can be stored in one byte. Examples are the letters of the alphabet.

character codes The single byte numeric codes that identify each ASCII code. These codes may differ from one computer to another. See ASCII.

code The instructions used to write a program.

concatenation The combining of two or more strings of characters to form one larger string.

data A general term used to describe both alphabetic (string) information and numeric information.

debugging The art of finding and fixing bugs in a program.

decimal The number system that humans use. It has ten individual digits: 0, 1, 3, 4, 5, 6, 7, 8, 9.

digit A single numeric character.

edit To change a section of program code.

enter The command that generally tells the computer that you are finished entering information into it through the key-board. This command is executed on the EL-5500III by pressing the **ENTER** key.

EPROM A type of memory used to store information in that doesn't need constant electrical power to maintain. The letters stand for Eraseable Programmable Read Only Memory.

firmware A term used to describe operating instructions for the computer that are generally found in ROM.

flowchart A method of outlining the logic of a program using diagrams.

hardware Computer equipment as opposed to computer programs. See software.

hexadecimal Also called **hex** for short. A number system based on sixteen different digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Each hex digit

represents for binary bits.

integer A positive or negative number which has no fractional part.

integrated circuit Also known as a "chip" or an "IC." It is a type of circuit that is very, very small. IC's are used to replace larger and generally more expensive circuit components.

kilobyte (K) A unit of computer memory which represents 1,024 bytes. Casually used to describe 1000 byte chunks of memory.

language A set of commands and statements that allow humans to communicate with computers. The EL-5500III uses BASIC.

laser Light Amplification through Stimulated Emission of Radiation. Lasers are a type of very concentrated and controllable light sources that can be used to do everything from welding a car to micro-surgery.

literal A value, either numeric or string, that is part of a program's code. As opposed to a variable.

loop A section of program code that does the same thing over and over.

machine code A method of programming a computer using binary, hexadecimal, octal, or decimal numbers to represent instructions. Very tedious but very powerful.

mass storage device This phrase is used to describe any number of pieces of machinery that can be used by a computer to store extremely large quantities of information. Some of the most common types are magnetic tapes and disks.

memory Electronic circuits in a computer that can be used to store information. Memory can be in many different forms: RAM, ROM, PROM, and EPROM.

nesting The practice of having one loop inside another loop.

number A character or set of characters that can be used as arithmetic operands in an equation or calculation.

octal A number system based on eight individual digits: 0, 1, 2, 3, 4, 5, 6, 7. Used instead of hex by some computer systems to represent groups of binary bits. Each octal digit represents three binary bits.

program The set of instruction written by the programmer that teaches the computer what to do.

PROM Programmable Read Only Memory.

RAM The section of computer memory that is used to write programs and to store data in the form of variables. The letters stand for Random Access Memory.

ROM The section of memory used by the computer to contain the BASIC interpreter language instructions. It is neither changeable nor usable by the programmer. The letters stand for Read Only Memory.

routine A program or set of instructions for a computer to execute.

scientific notation A method of representing decimal numbers in powers of ten. It enables the computer to store very large numbers in a very small amount of memory.

software Computer programs.

statement An instruction to the computer in a program.

string Characters that cannot be used as numeric operands. See **number**.

subroutine A section of a program that is referenced or called by another section of the same program.

syntax The proper way in which to structure an instruction to the computer so that the BASIC interpreter can understand the instruction and carry it out.

INDEX

A

B

C

F

G

H

I

L

COMPUTING THE SCIENCES

Printed in Japan
8F3T(TMANE1003ECZZ)⑤